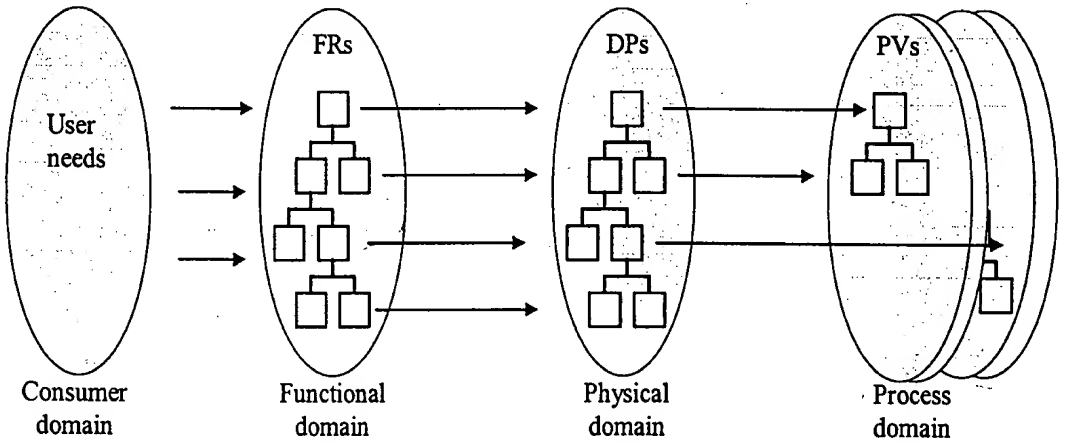


FIGURE 1

Uncoupled - *Desired*

	DP1	DP2	DP3
FR1	X	O	O
FR2	O	X	O
FR3	O	O	X

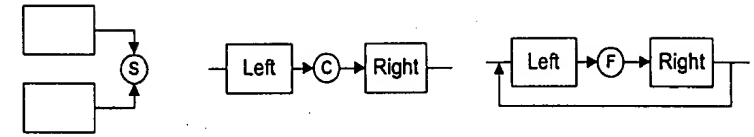
Decoupled - *Acceptable*

	DP1	DP2	DP3
FR1	X	O	O
FR2	X	X	O
FR3	X	X	X

Coupled - *Undesired*

	DP1	DP2	DP3
FR1	X	X	X
FR2	X	X	X
FR3	X	X	X

FIGURE 2



Summation Junction (Uncoupled Design)

Control Junction (Decoupled Design)

Feedbaek Junction (Coupled Design)

FIGURE 3

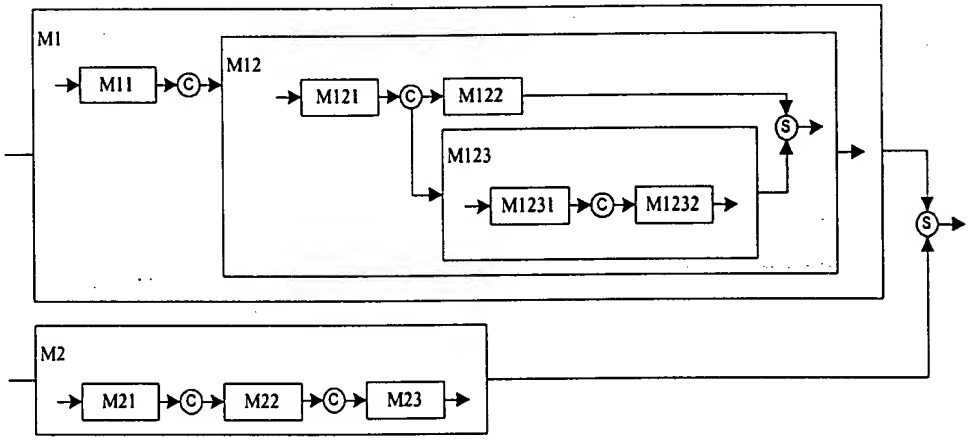


FIGURE 4

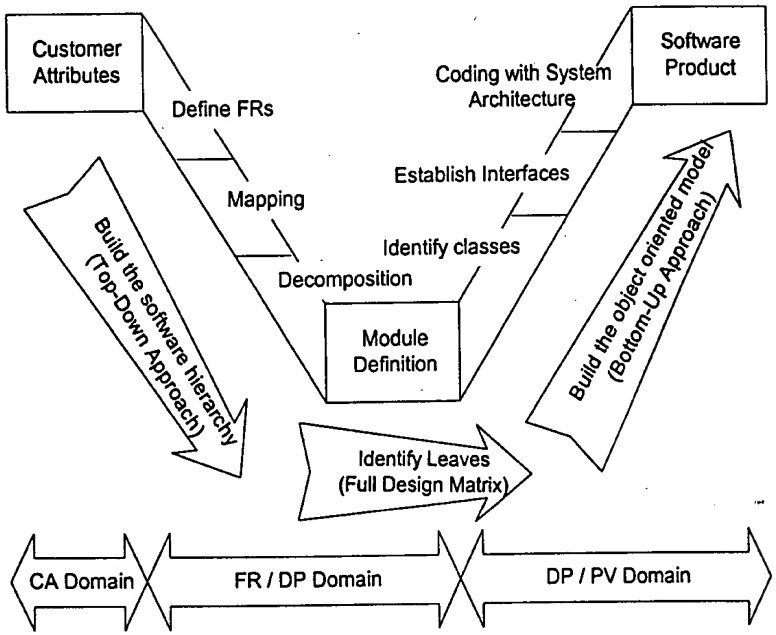


FIGURE 5

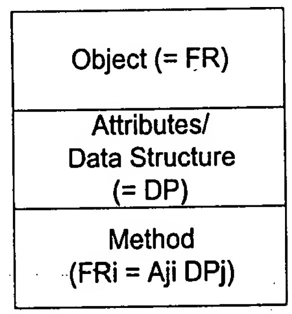


FIGURE 6

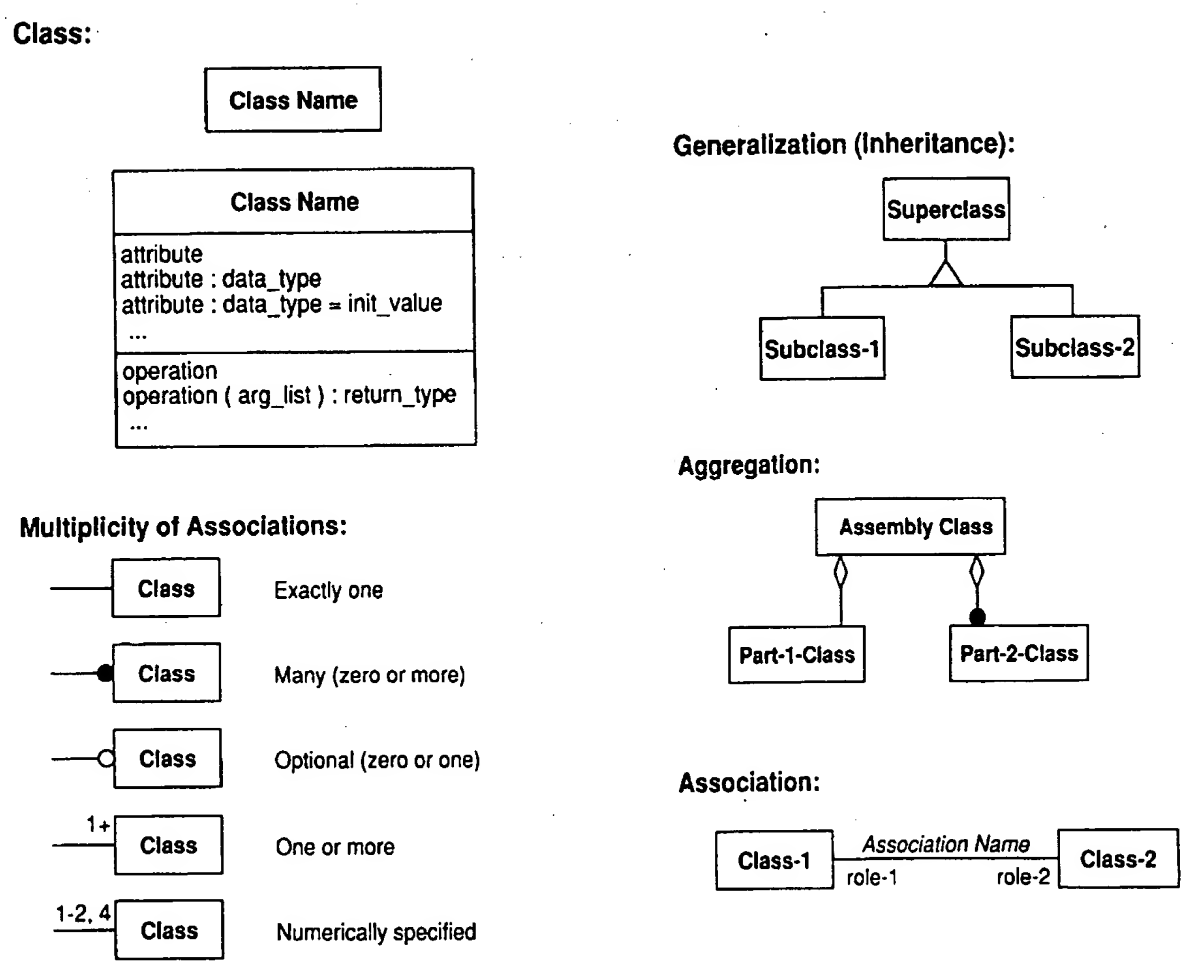

Class:
Class Name
Class Name
attribute
attribute : data_type
attribute : data_type = init_value
...
operation
operation (arg_list) : return_type
...
Multiplicity of Associations:
Class
Exactly one
Class
Many (zero or more)
Class
Optional (zero or one)
1+
Class
One or more
1-2, 4
Class
Numerically specified
Generalization (Inheritance):
Superclass
Subclass-1
Subclass-2
Aggregation:
Assembly Class
Part-1-Class
Part-2-Class
Association:
Class-1
Association Name
role-1
role-2
Class-2

FIGURE 7

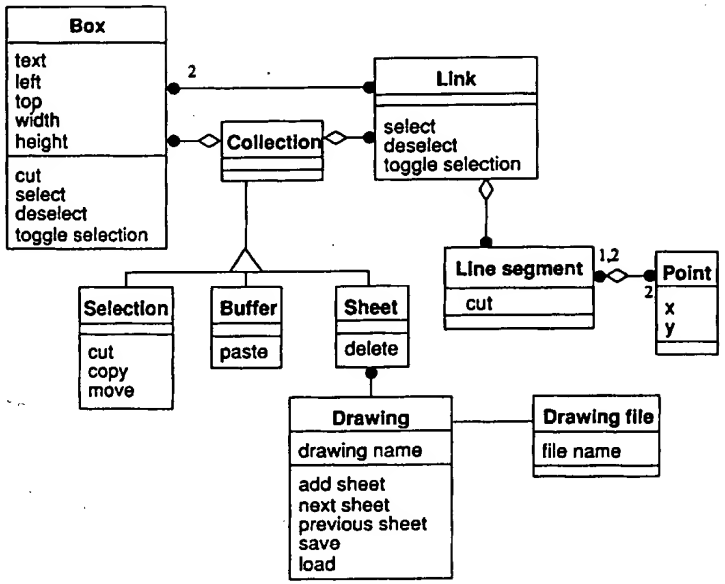
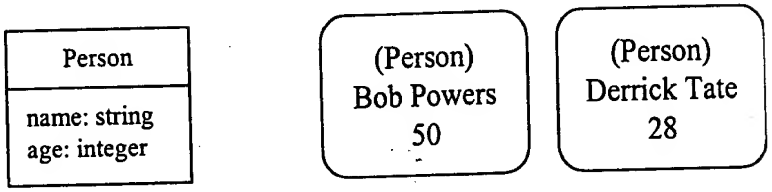


FIGURE 8



Class Diagram

Instance Diagram

FIGURE 9

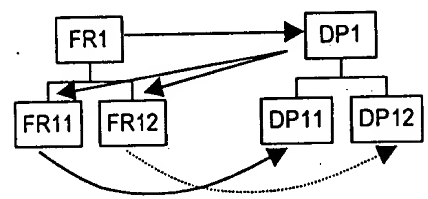


FIGURE 10

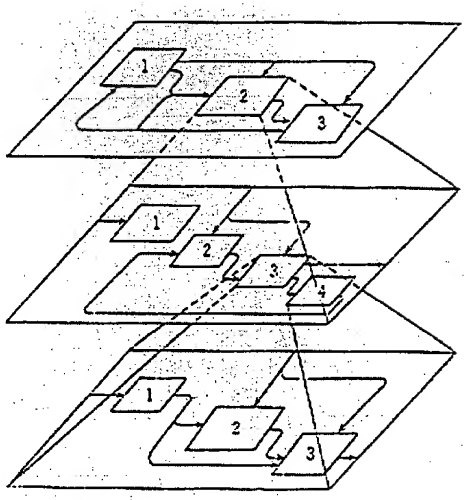


FIGURE 11

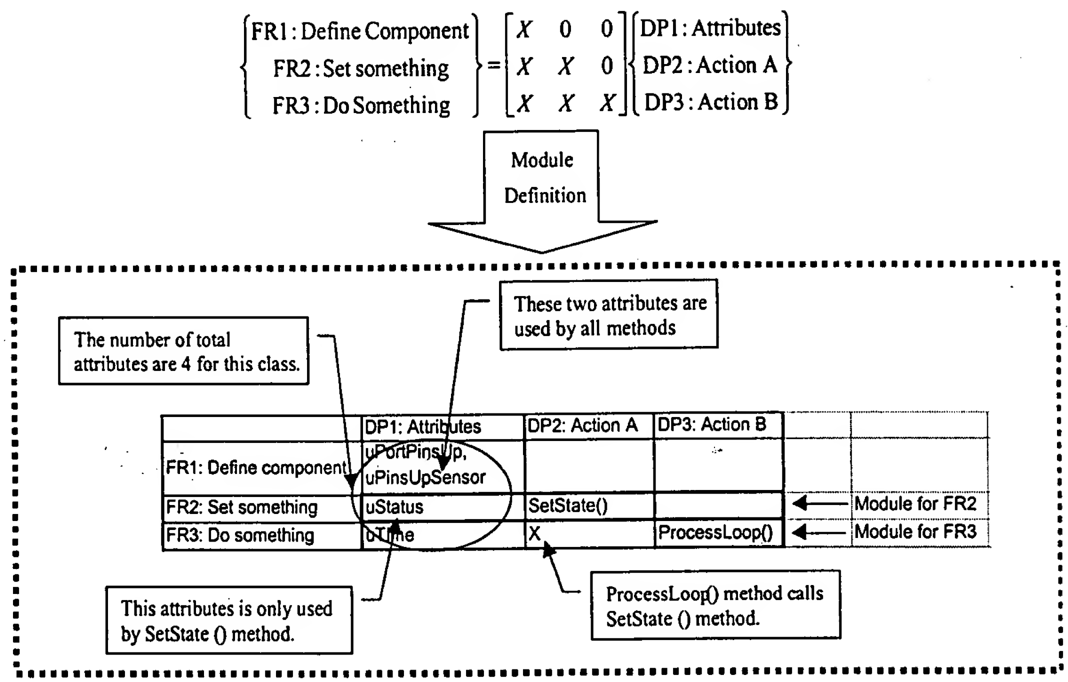

FR1 : Define Component
FR2 : Set something
FR3 : Do Something
X 0 0
X X 0
X X X
DP1 : Attributes
DP2 : Action A
DP3 : Action B
Module Definition
The number of total attributes are 4 for this class.
These two attributes are used by all methods
DP1: Attributes
DP2: Action A
DP3: Action B
FR1: Define component
uPortPinsUp, uPinsUpSensor
FR2: Set something
uStatus
SetState()
Module for FR2
FR3: Do something
uTime
X
ProcessLoop()
Module for FR3
This attributes is only used by SetState () method.
ProcessLoop() method calls SetState () method.

FIGURE 12

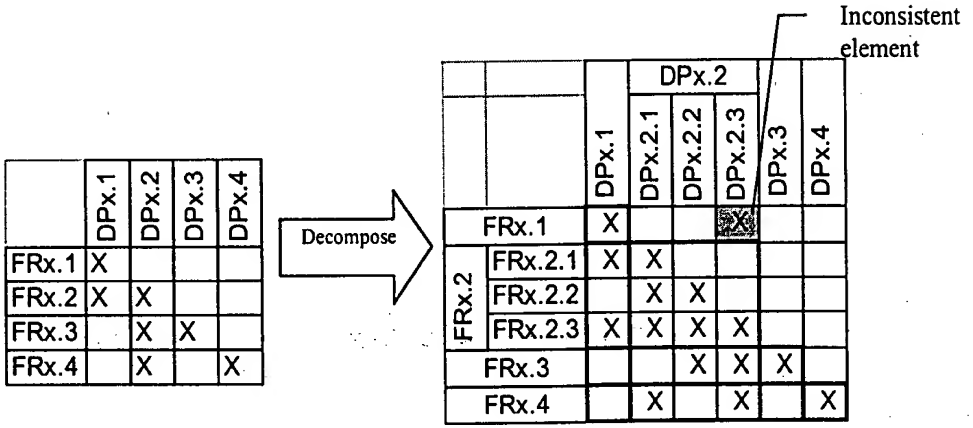


FIGURE 13

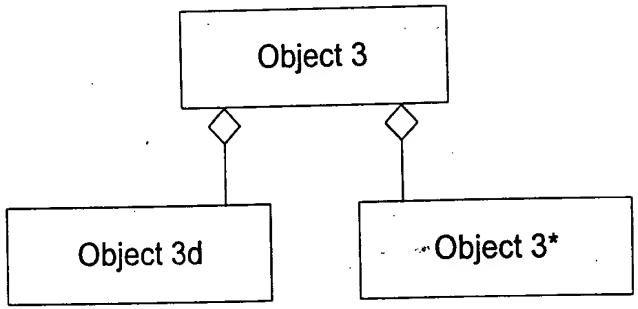
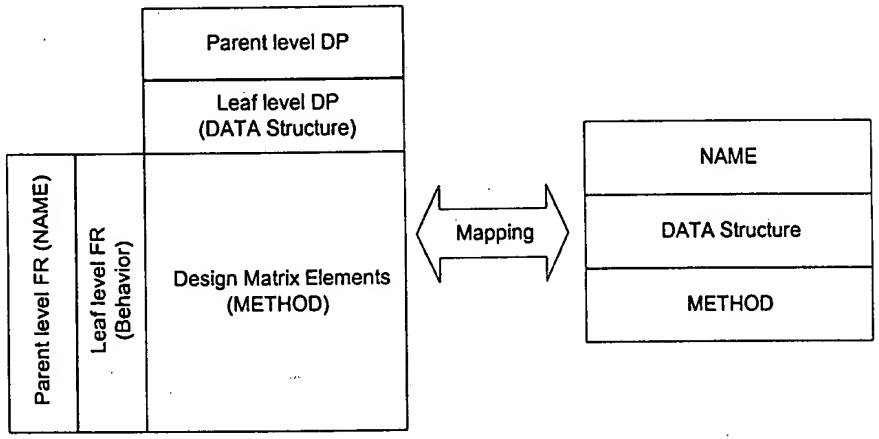


FIGURE 14



(a) Full Design Matrix Table

(b) Class Diagram

FIGURE 15

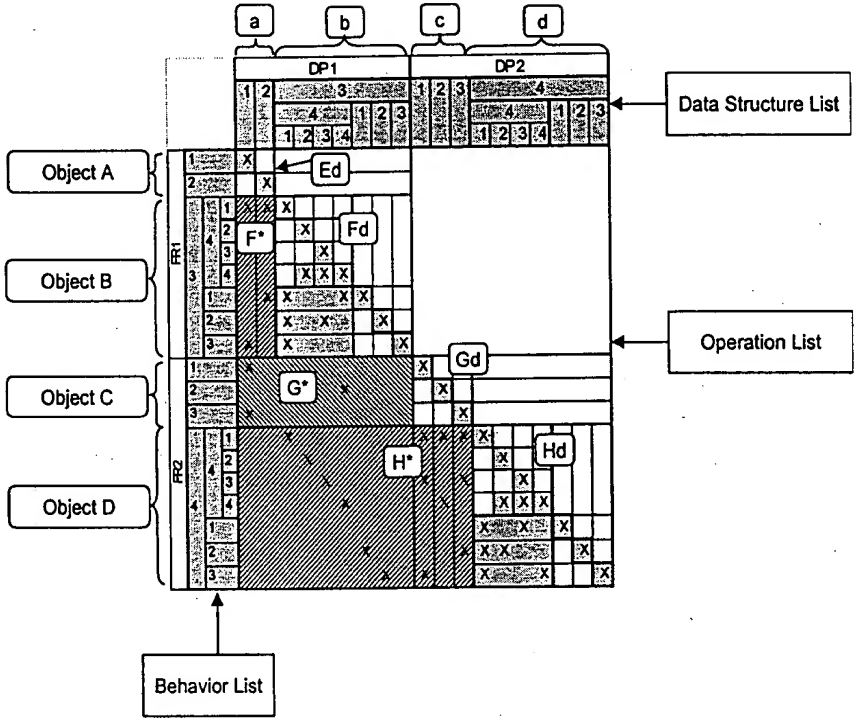


FIGURE 16

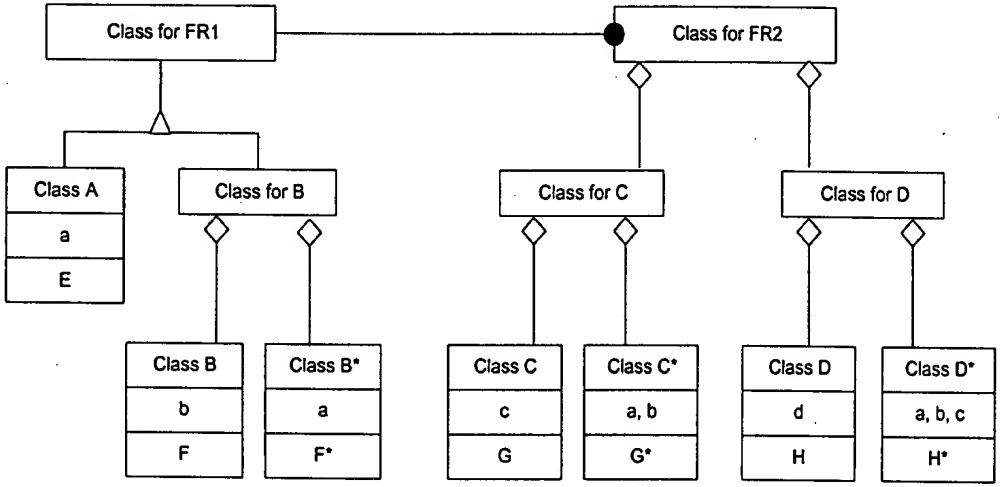

Class for FR1
Class for FR2
Class A
a
E
Class for B
Class for C
Class for D
Class B
b
F
Class B*
a
F*
Class C
c
G
Class C*
a, b
G*
Class D
d
H
Class D*
a, b, c
H*

FIGURE 17

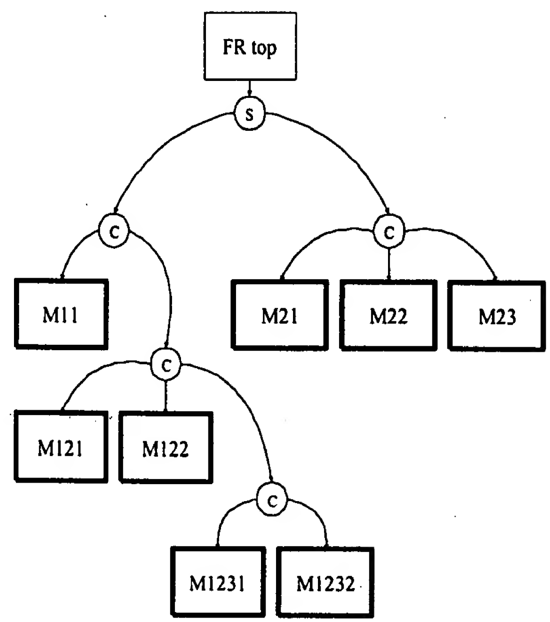


FIGURE 18

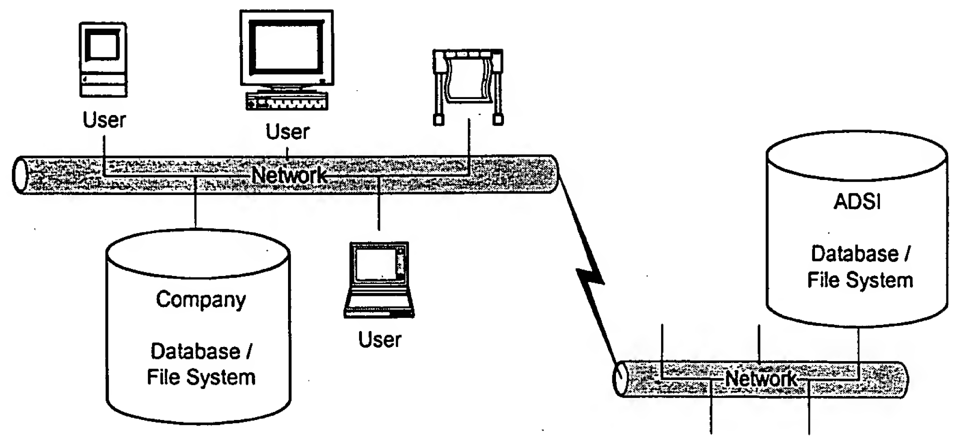

User
User
Network
Company
Database / File System
User
ADSI
Database / File System
Network

FIGURE 19

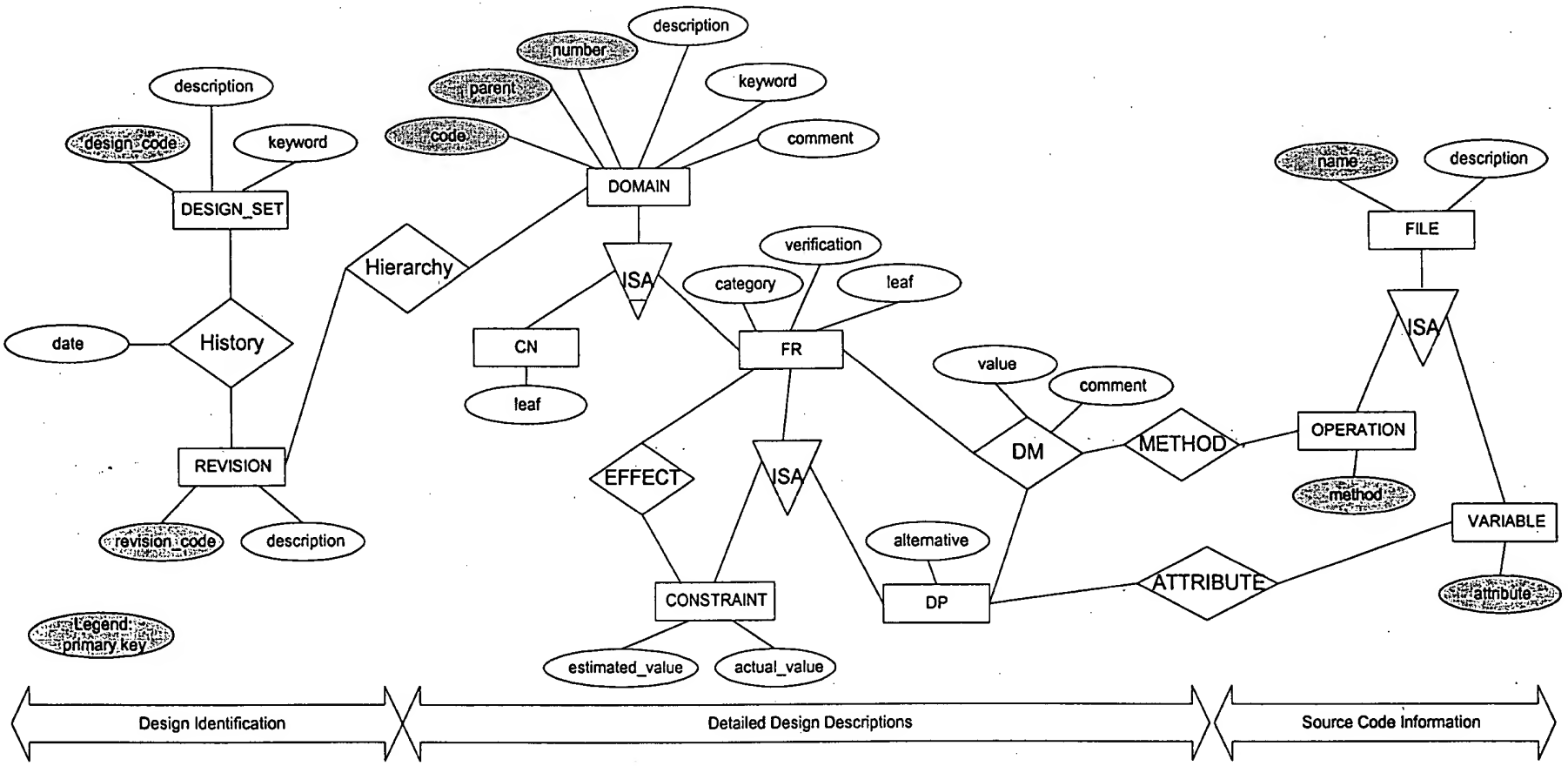


FIGURE 20

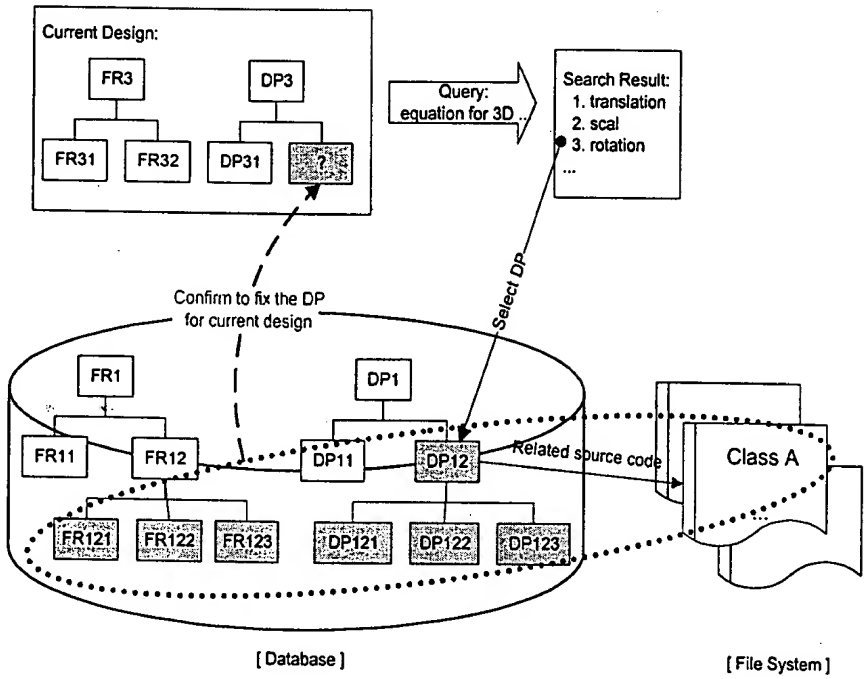


FIGURE 21

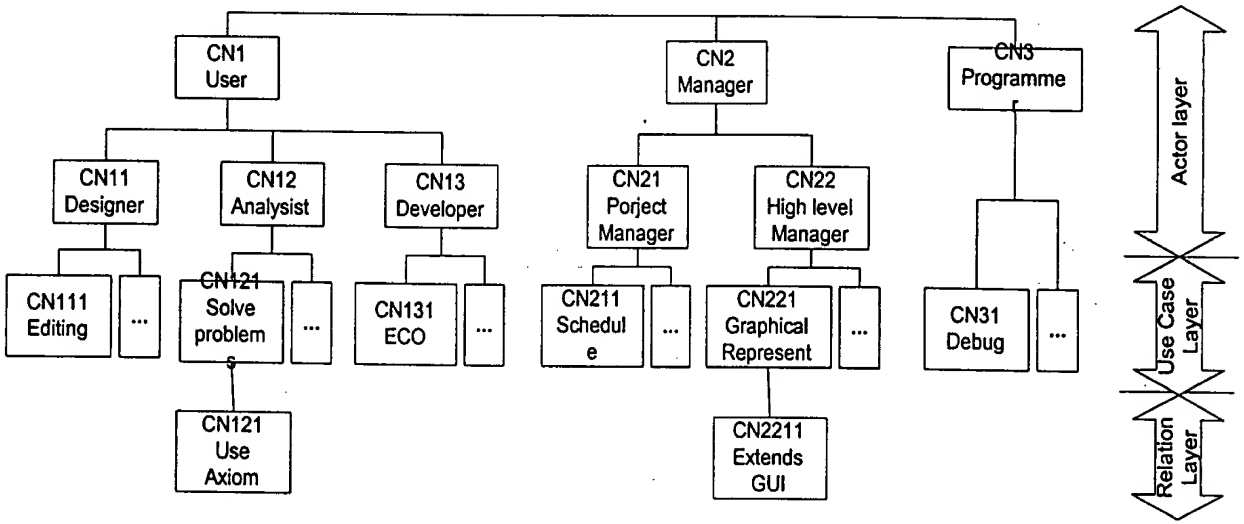

CN1
User
CN2
Manager
CN3
Programme
r
CN11
Designer
CN12
Analysist
CN13
Developer
CN21
Porject
Manager
CN22
High level
Manager
CN111
Editing
...
CN121
Solve
problem
s
...
CN131
ECO
...
CN211
Schedul
e
...
CN221
Graphical
Represent
...
CN31
Debug
...
CN121
Use
Axiom
CN2211
Extends
GUI
Actor layer
Use Case
Layer
Relation
Layer

FIGURE 22

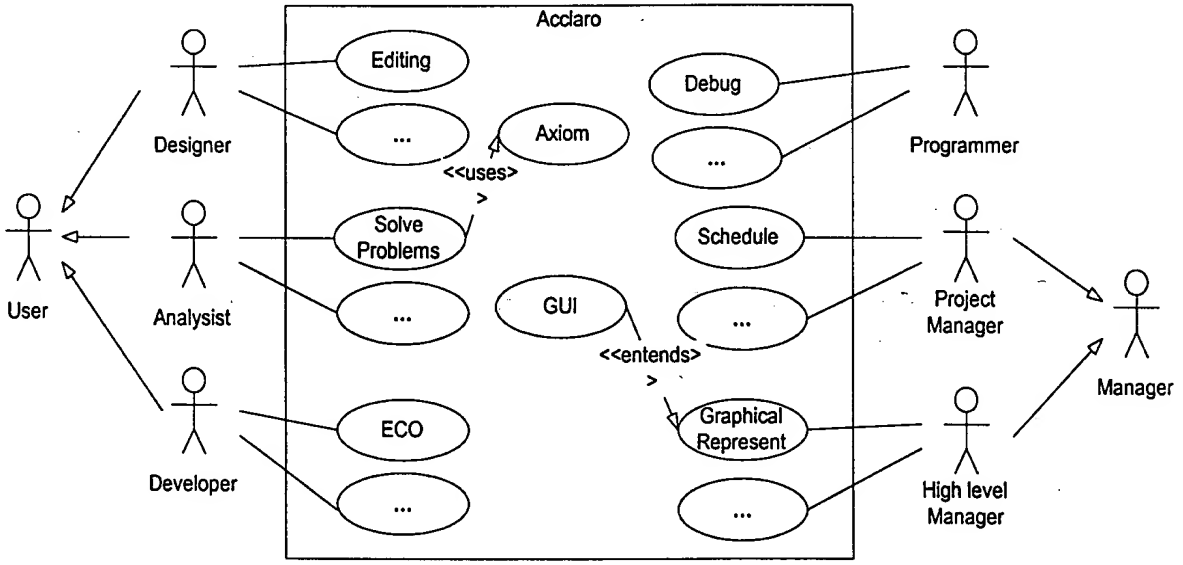


FIGURE 23

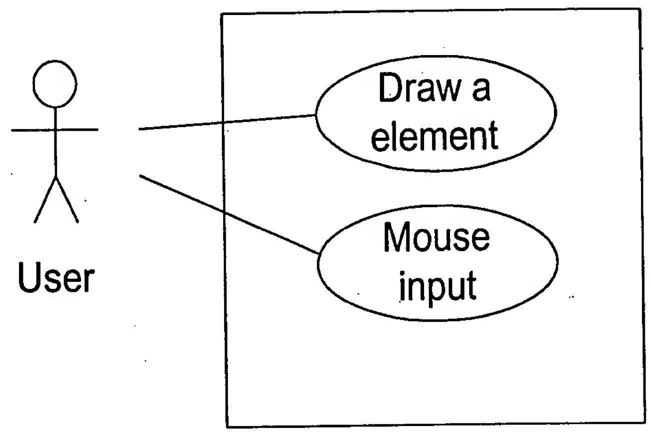


FIGURE 24

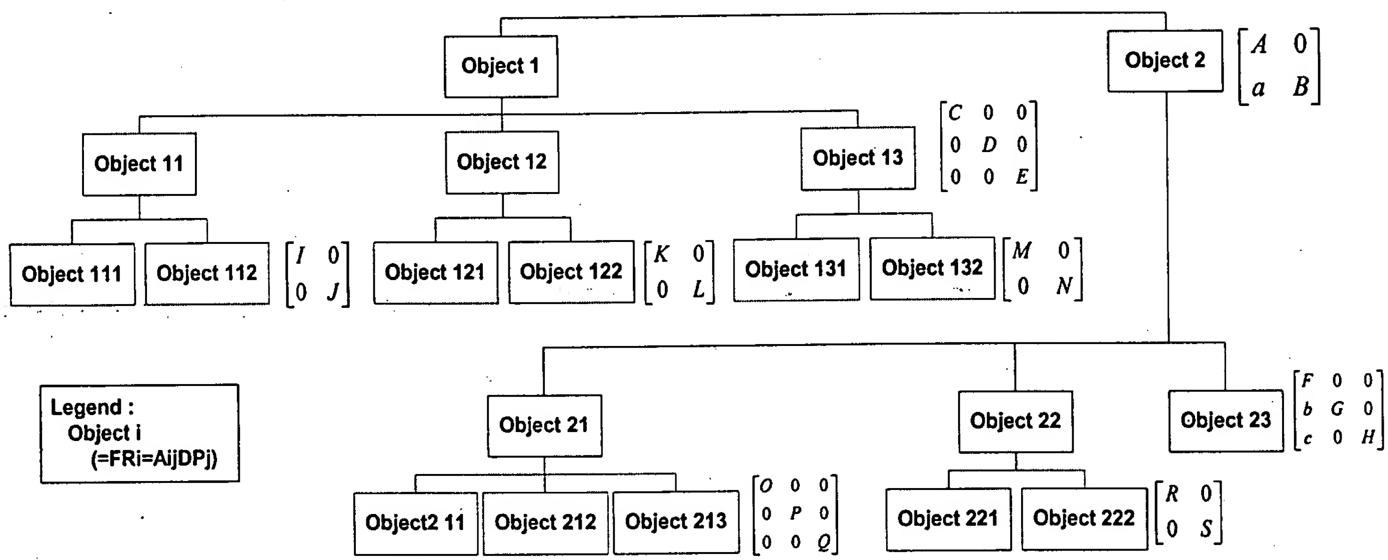

Object 1
Object 2
Object 11
Object 12
Object 13
Object 111
Object 112
Object 121
Object 122
Object 131
Object 132
Object 21
Object 22
Object 23
Object2 11
Object 212
Object 213
Object 221
Object 222
Legend :
Object i
(=FRi=AijDPj)

FIGURE 25

On-diagonal element for the intermediate or higher level

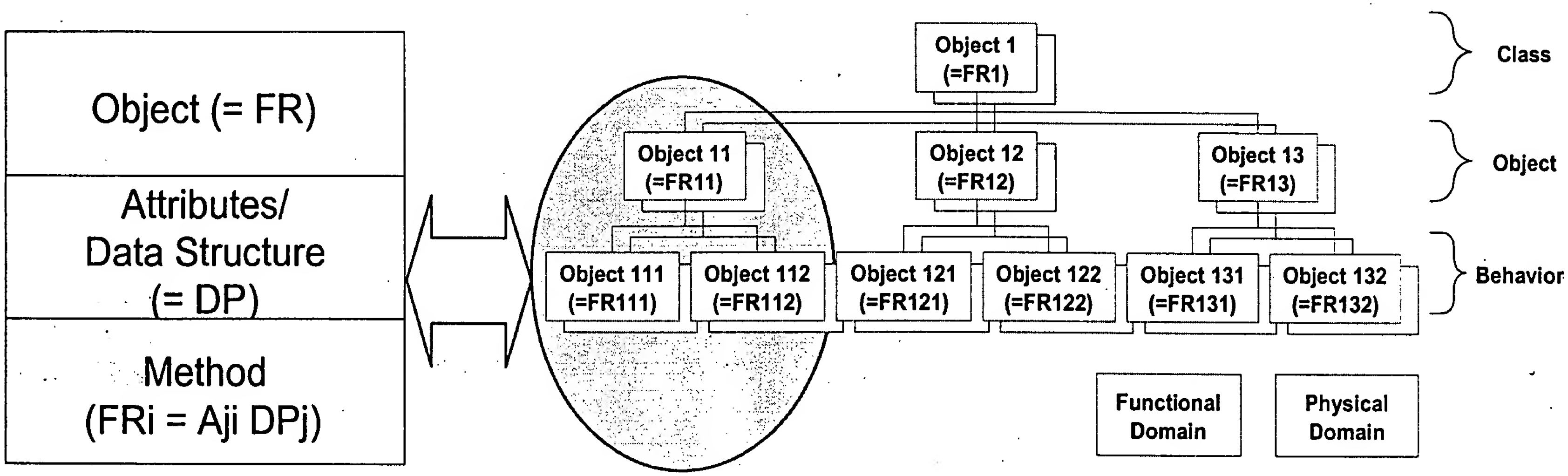
Off-diagonal element for the intermediate or higher level

Off-diagonal element for the leaf or lower level

			DP1: Element characteristics						DP2: GUI with window					
			DP11: Line characteristics		DP12: Rectangle characteristic		DP13: Circle characteristic		DP21: Radio buttons			DP22: Mouse click information		
			DP111: Start point	DP112: End point	DP121: Upper left point	DP122: Lower right point	DP131: Center point	dP132: Radius	DP211: Line button	DP212: Rectangle button	DP213: Circle button	DP221: Event for push	DP222: Event for release	DP23: Drawing area
FR1: Define element	FR11: Define line element	FR111: Define start	I											
		FR112: Define end		J										
	FR12: Define rectangle element	FR121: Define upper left corner			K									
		FR122: Define lower right corner				L								
	FR13: Define circle element	FR131: Define center					M							
		FR132: Define radius						N						
FR2: Specify drawing	FR21: Identify the drawing type	FR211: Identify line							O					
		FR212: Identify rectangle								P				
		FR213: Identify circle									Q			
	FR22: Detect drawing location	FR221: Detect mouse push	X		X		X		X	X	X	R		
		FR222: Detect mouse release		X		X		X	X	X	X		S	
	FR23: Draw the element		X	X	X	X	X	X	X	X	X			H

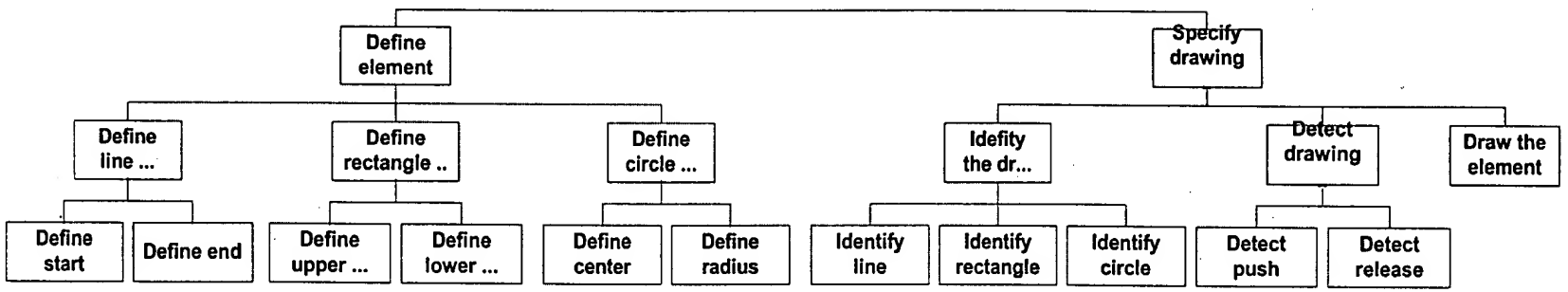
A B C D E F G a b c

FIGURE 26

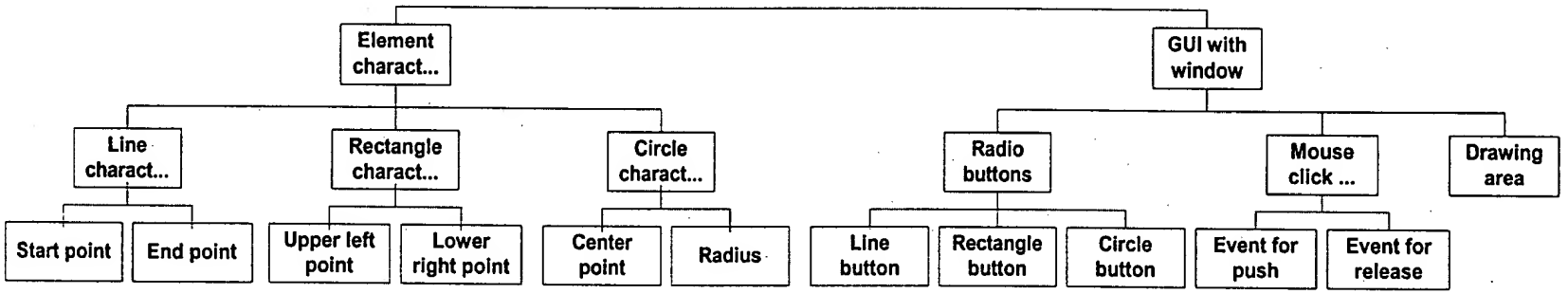


Object (= FR)
Attributes/
Data Structure
(= DP)
Method
(FRi = Aji DPj)
Object 1
(=FR1)
Object 11
(=FR11)
Object 12
(=FR12)
Object 13
(=FR13)
Object 111
(=FR111)
Object 112
(=FR112)
Object 121
(=FR121)
Object 122
(=FR122)
Object 131
(=FR131)
Object 132
(=FR132)
Class
Object
Behavior
Functional
Domain
Physical
Domain

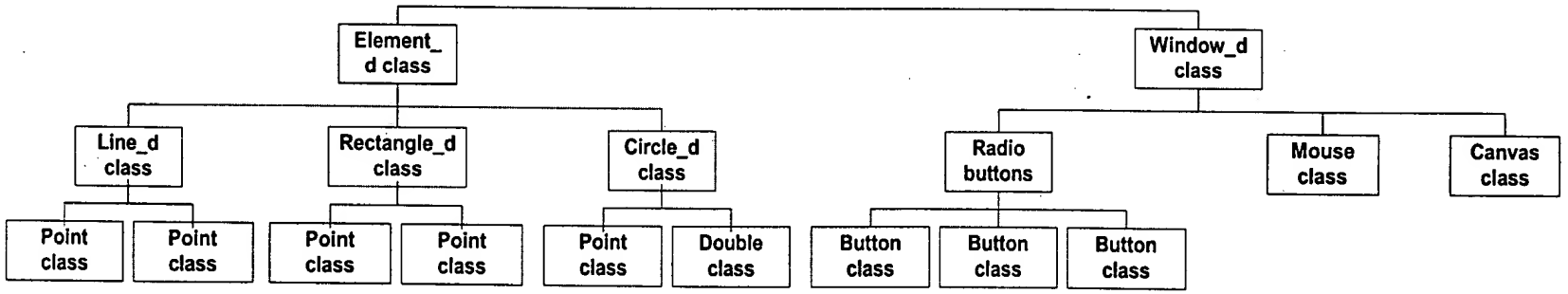
FIGURE 27



(a) Functional Domain



(b) Physical Domain



(c) Process Domain

FIGURE 28

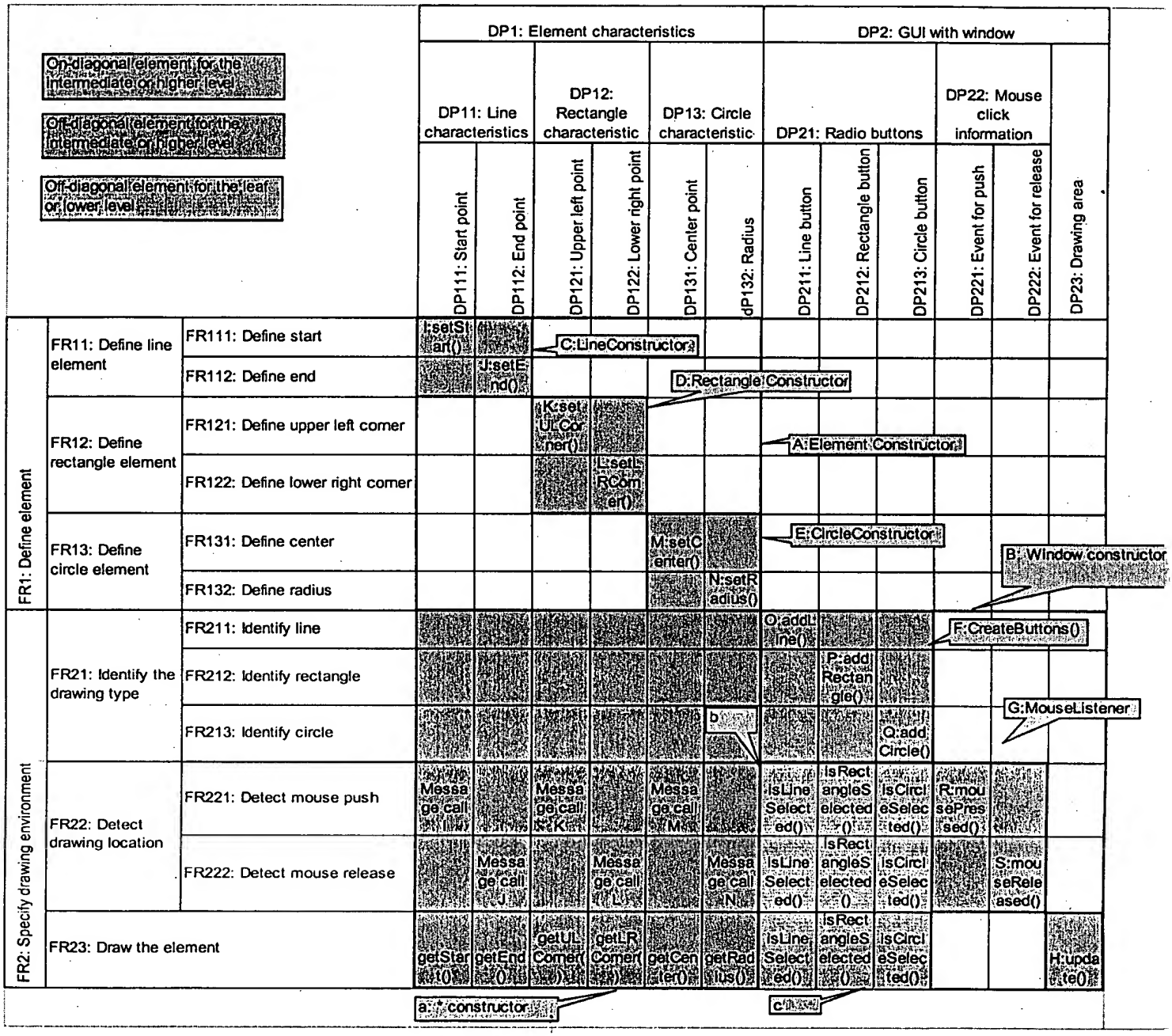

On-diagonal element for the intermediate or higher level
Off-diagonal element for the intermediate or higher level
Off-diagonal element for the leaf or lower level
DP1: Element characteristics
DP2: GUI with window
DP11: Line characteristics
DP12: Rectangle characteristic
DP13: Circle characteristic
DP21: Radio buttons
DP22: Mouse click information
DP111: Start point
DP112: End point
DP121: Upper left point
DP122: Lower right point
DP131: Center point
dP132: Radius
DP211: Line button
DP212: Rectangle button
DP213: Circle button
DP221: Event for push
DP222: Event for release
DP23: Drawing area
FR1: Define element
FR11: Define line element
FR111: Define start
FR112: Define end
FR12: Define rectangle element
FR121: Define upper left corner
FR122: Define lower right corner
FR13: Define circle element
FR131: Define center
FR132: Define radius
FR2: Specify drawing environment
FR21: Identify the drawing type
FR211: Identify line
FR212: Identify rectangle
FR213: Identify circle
FR22: Detect drawing location
FR221: Detect mouse push
FR222: Detect mouse release
FR23: Draw the element
I:setStart()
J:setEnd()
C:LineConstructor
D:RectangleConstructor
K:setULCorner()
L:setLRCorner()
A:ElementConstructor
M:setCenter()
N:setRadius()
E:CircleConstructor
B:WindowConstructor
O:addLine()
P:addRectangle()
Q:addCircle()
F:CreateButtons()
G:MouseListener
b
Message call
isLineSelected()
isRectangleSelected()
isCircleSelected()
R:mousePressed()
S:mouseReleased()
getStart()
getEnd()
getULCorner()
getLRCorner()
getCenter()
getRadius()
isLineSelected()
isRectangleSelected()
isCircleSelected()
H:update()
a: constructor
c

FIGURE 29

Object	Object 111/112/121/122/131	Object 132	Object 11		Object 12		Object 13		Object 1		Object 2		Object 211/212/213	Object 22	Object 23	Object 1*	
Name	Point	Double	Line_d		Rectangle_d		Circle_d		Element_d		Window_d		Radio B	Mouse	Canvas	Element_*	
Attribute			DP111	Point start	DP121	Point upper_left	DP131	Point center	DP11	Line l	DP211	Radiobutton line					
			DP112	Point end	DP122	Point lower_right	DP132	Double radius	DP12	Rectangle r	DP212	Radiobutton rectangle					
									DP13	Circle c	DP213	Radiobutton circle					
											DP22	Mouse m					
											DP23	Canvas c					
Method			C	Line()	D	Rectangle()	E	Center()	A	Element()	B	Window()				a	Element*()
			I	setStart()	K	setULComer()	M	setCenter()			F	CreateButtons()					getStart()
			J	setEnd()	L	setLRComer()	N	setRadius()			O	addLine()					getEnd()
											P	addRectangle					getULComer()
											Q	addCircle()					getLRComer()
											G	implement MouseLisner					getCenter()
											R	mousePresed()					getRadius()
											S	mouseReleased()					assignLine()
											H	draw()					assignRectangle()
											b/c	isLineSelected()					assignCircle()
											b/c	isRectangleSelected()					
											b/c	isCircleSelected()					

FIGURE 30

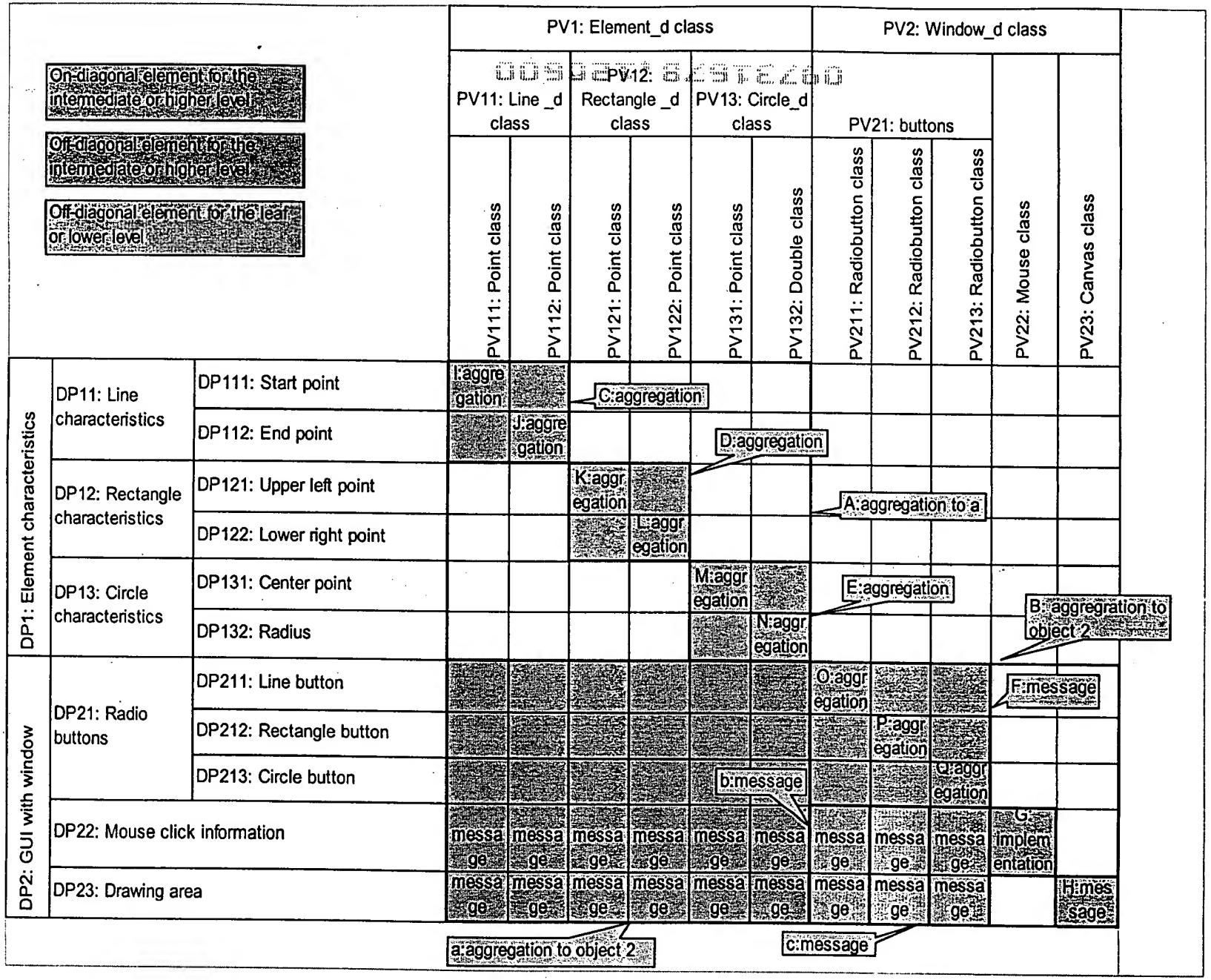
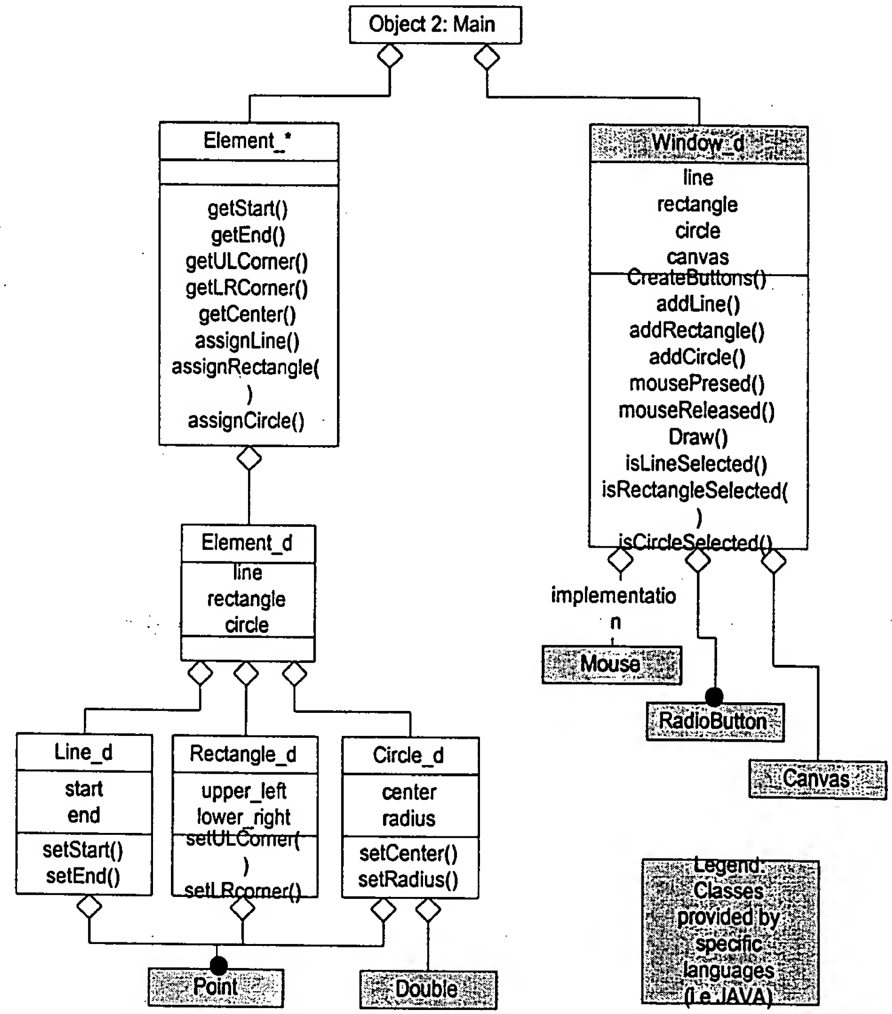

On-diagonal element for the intermediate or higher level
Off-diagonal element for the intermediate or higher level
Off-diagonal element for the leaf or lower level
PV1: Element_d class
PV2: Window_d class
PV11: Line _d class
PV12: Rectangle _d class
PV13: Circle_d class
PV21: buttons
PV111: Point class
PV112: Point class
PV121: Point class
PV122: Point class
PV131: Point class
PV132: Double class
PV211: Radiobutton class
PV212: Radiobutton class
PV213: Radiobutton class
PV22: Mouse class
PV23: Canvas class
DP1: Element characteristics
DP11: Line characteristics
DP111: Start point
DP112: End point
DP12: Rectangle characteristics
DP121: Upper left point
DP122: Lower right point
DP13: Circle characteristics
DP131: Center point
DP132: Radius
DP2: GUI with window
DP21: Radio buttons
DP211: Line button
DP212: Rectangle button
DP213: Circle button
DP22: Mouse click information
DP23: Drawing area
I:aggregation
C:aggregation
J:aggregation
D:aggregation
K:aggregation
A:aggregation to a
L:aggregation
M:aggregation
E:aggregation
B: aggregration to object 2
N:aggregation
O:aggregation
F:message
P:aggregation
Q:aggregation
b:message
message
G: implementation
H:message
a:aggregation to object 2
c:message

FIGURE 31



Object 2: Main
Element_*
getStart()
getEnd()
getULCorner()
getLRCorner()
getCenter()
assignLine()
assignRectangle(
)
assignCircle()
Window_d
line
rectangle
circle
canvas
CreateButtons()
addLine()
addRectangle()
addCircle()
mousePresed()
mouseReleased()
Draw()
isLineSelected()
isRectangleSelected(
)
isCircleSelected()
implementatio
n
Mouse
RadioButton
Canvas
Element_d
line
rectangle
circle
Line_d
start
end
setStart()
setEnd()
Rectangle_d
upper_left
lower_right
setULCorner(
)
setLRcorner()
Circle_d
center
radius
setCenter()
setRadius()
Point
Double
Legend:
Classes
provided by
specific
languages
(i.e.JAVA)

FIGURE 32

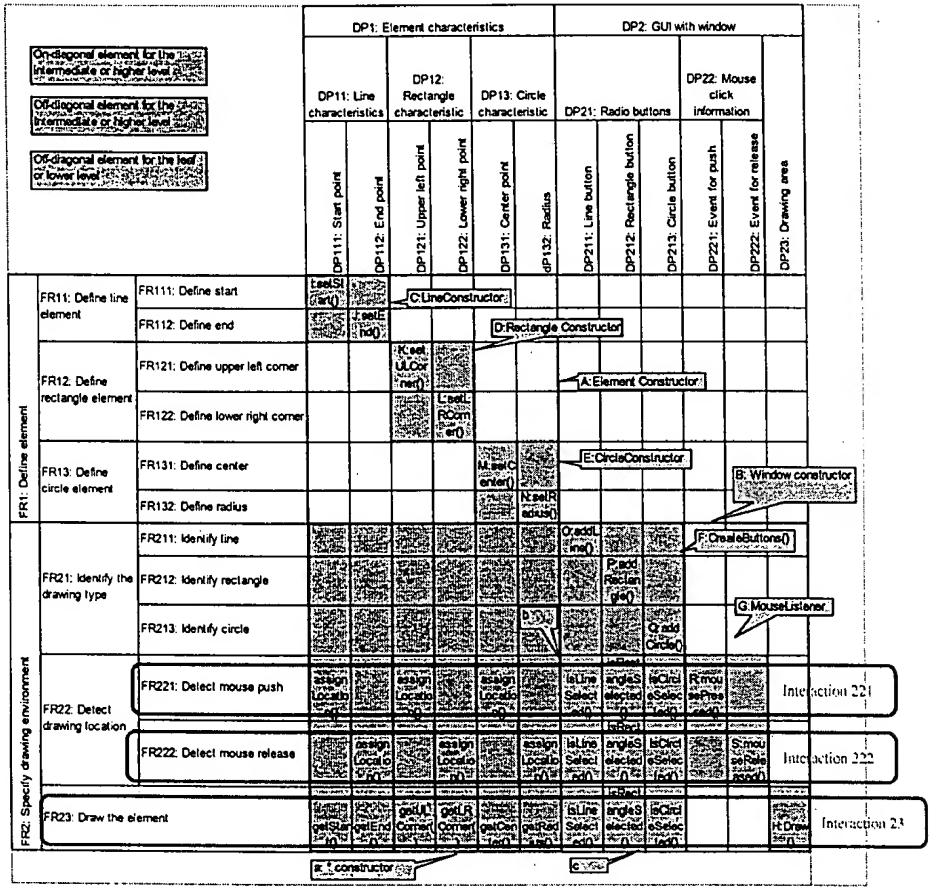

On-diagonal element for the intermediate or higher level
Off-diagonal element for the intermediate or higher level
Off-diagonal element for the leaf or lower level
DP1: Element characteristics
DP2: GUI with window
DP11: Line characteristics
DP12: Rectangle characteristic
DP13: Circle characteristic
DP21: Radio buttons
DP22: Mouse click information
DP111: Start point
DP112: End point
DP121: Upper left point
DP122: Lower right point
DP131: Center point
dP132: Radius
DP211: Line button
DP212: Rectangle button
DP213: Circle button
DP221: Event for push
DP222: Event for release
DP23: Drawing area
FR1: Define element
FR11: Define line element
FR111: Define start
FR112: Define end
FR12: Define rectangle element
FR121: Define upper left corner
FR122: Define lower right corner
FR13: Define circle element
FR131: Define center
FR132: Define radius
FR2: Specify drawing environment
FR21: Identify the drawing type
FR211: Identify line
FR212: Identify rectangle
FR213: Identify circle
FR22: Detect drawing location
FR221: Detect mouse push
FR222: Detect mouse release
FR23: Draw the element
C:LineConstructor
D:Rectangle Constructor
A:Element Constructor
E:CircleConstructor
B: Window constructor
F:CreateButtons()
G:MouseListener
Interaction 221
Interaction 222
Interaction 23
constructor
c

FIGURE 33

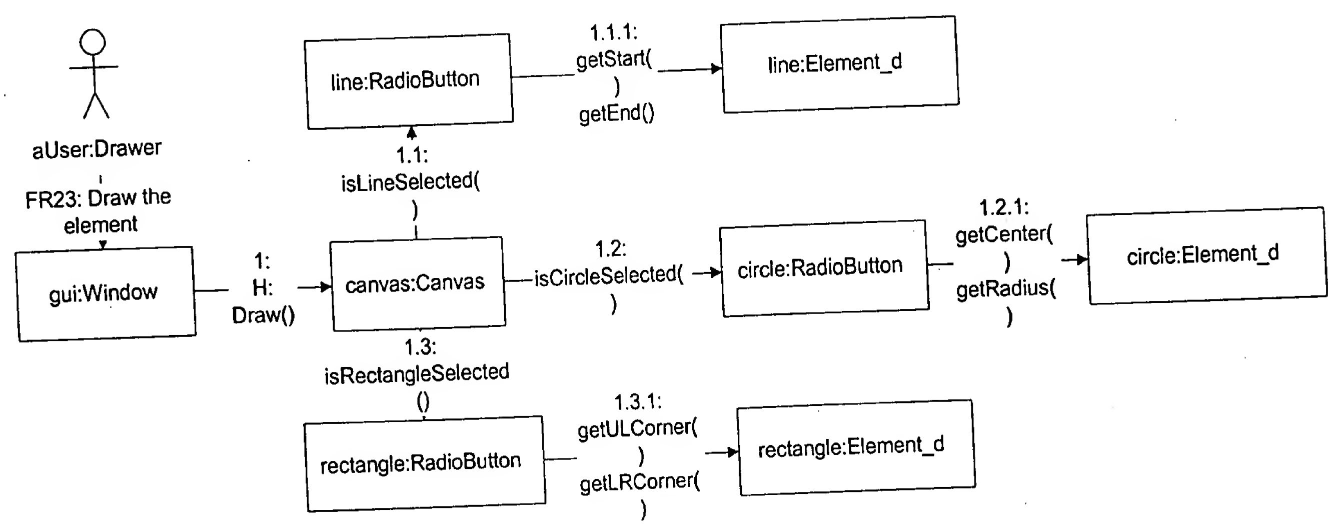
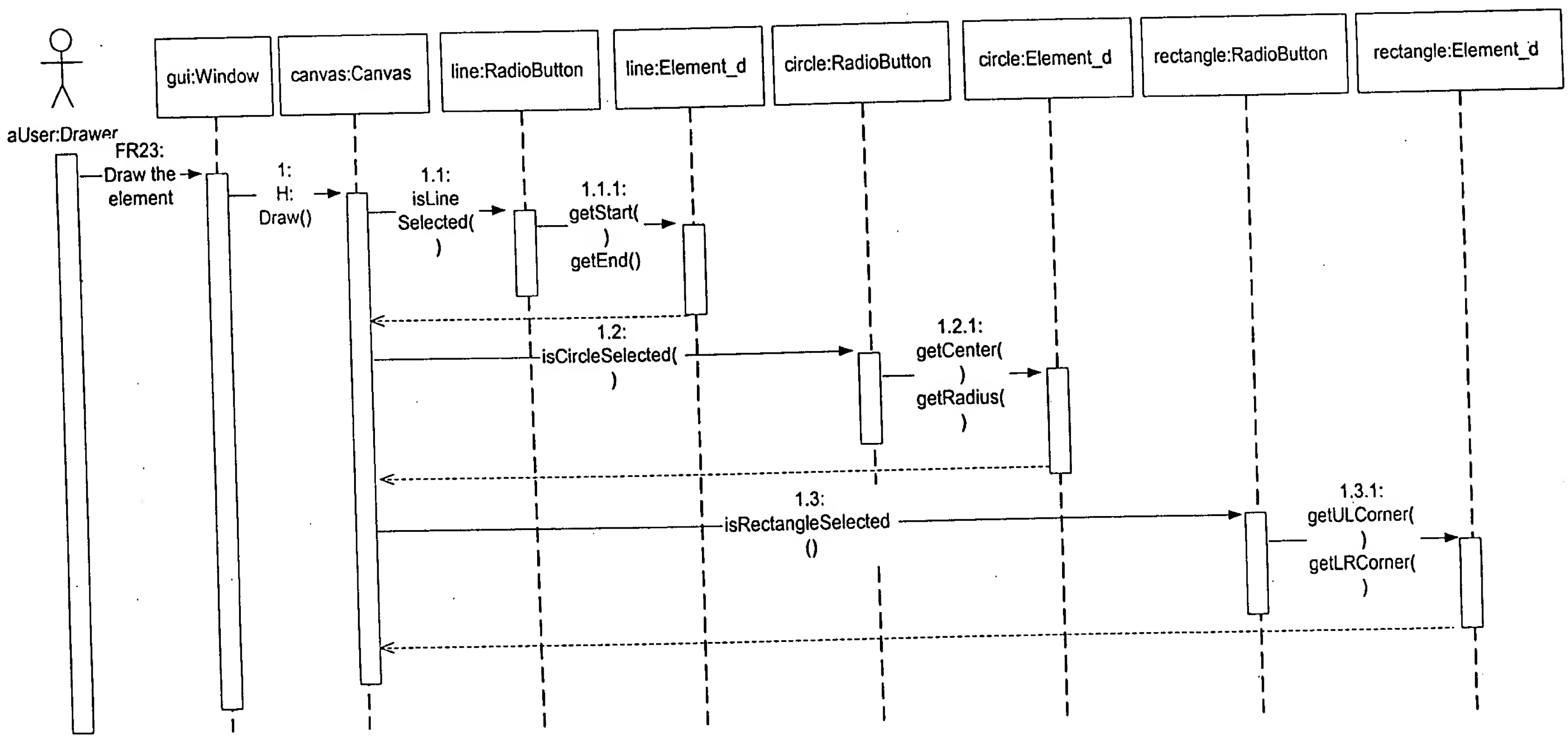

aUser:Drawer
FR23: Draw the element
gui:Window
1: H: Draw()
canvas:Canvas
1.1: isLineSelected( )
line:RadioButton
1.1.1: getStart( ) getEnd()
line:Element_d
1.2: isCircleSelected( )
circle:RadioButton
1.2.1: getCenter( ) getRadius( )
circle:Element_d
1.3: isRectangleSelected ()
rectangle:RadioButton
1.3.1: getULCorner( ) getLRCorner( )
rectangle:Element_d

FIGURE 34



aUser:Drawer
gui:Window
canvas:Canvas
line:RadioButton
line:Element_d
circle:RadioButton
circle:Element_d
rectangle:RadioButton
rectangle:Element_d
FR23: Draw the element
1: H: Draw()
1.1: isLine Selected( )
1.1.1: getStart( ) getEnd()
1.2: isCircleSelected( )
1.2.1: getCenter( ) getRadius( )
1.3: isRectangleSelected ()
1.3.1: getULCorner( ) getLRCorner( )

FIGURE 35

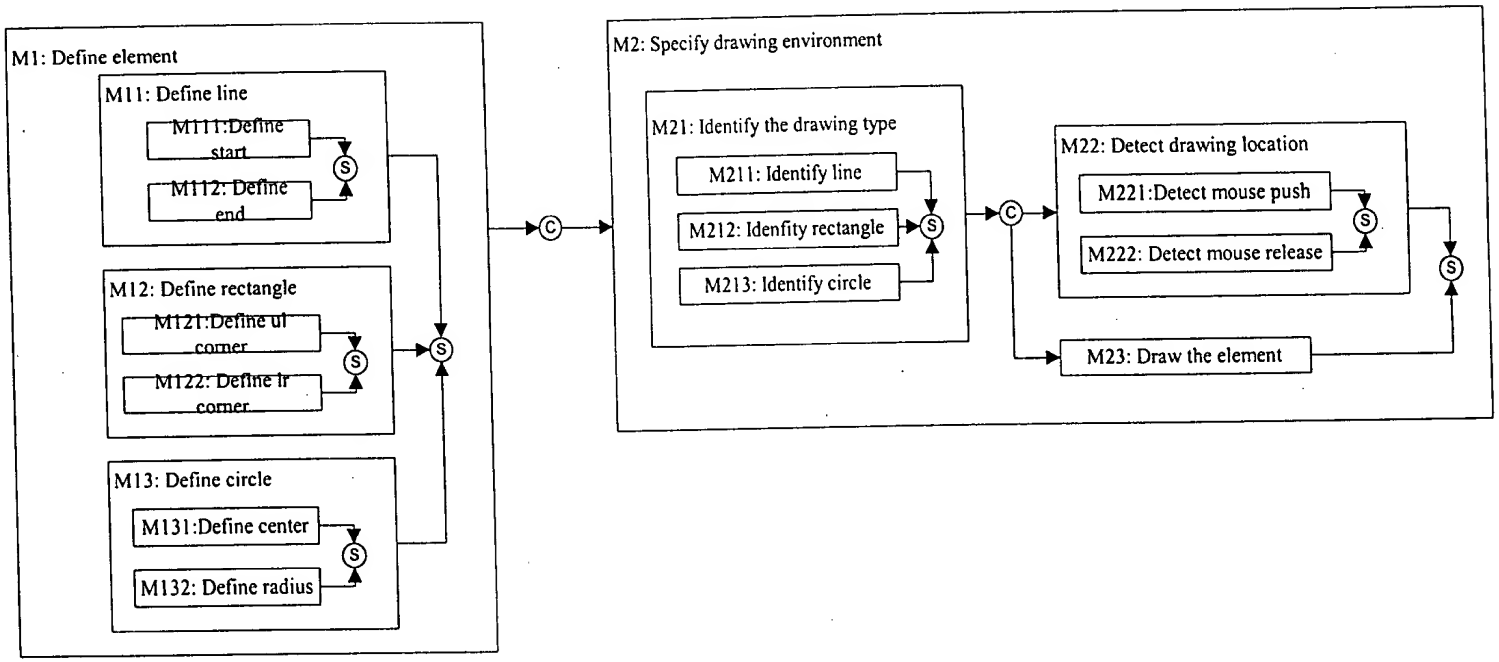


FIGURE 36

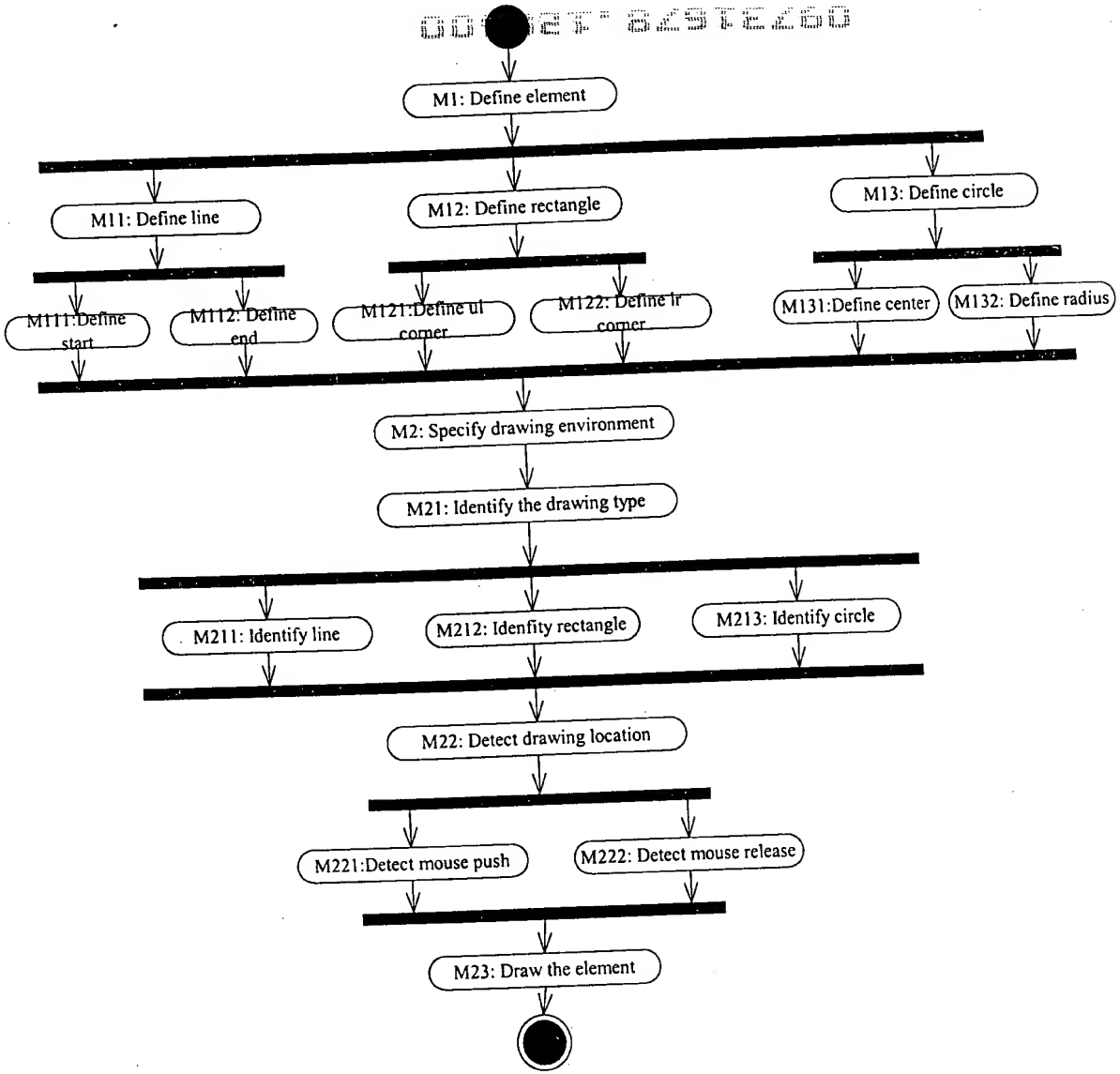

M1: Define element
M11: Define line
M12: Define rectangle
M13: Define circle
M111:Define start
M112: Define end
M121:Define ul corner
M122: Define lr corner
M131:Define center
M132: Define radius
M2: Specify drawing environment
M21: Identify the drawing type
M211: Identify line
M212: Idenfity rectangle
M213: Identify circle
M22: Detect drawing location
M221:Detect mouse push
M222: Detect mouse release
M23: Draw the element

FIGURE 37

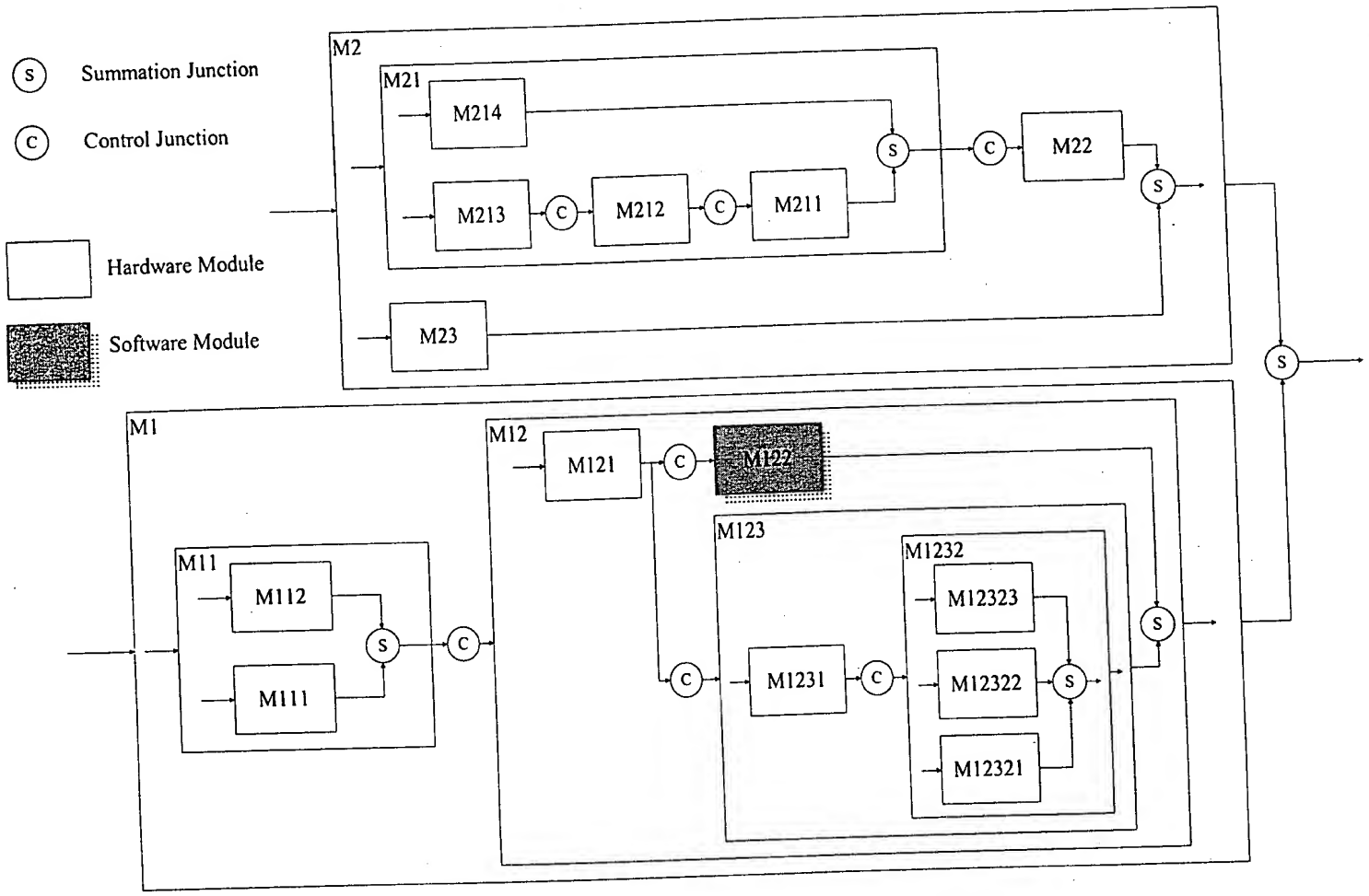

S Summation Junction
C Control Junction
Hardware Module
Software Module
M2
M21
M214
M213
M212
M211
M22
M23
M1
M11
M112
M111
M12
M121
M122
M123
M1232
M12323
M1231
M12322
M12321

FIGURE 38

```
(object Petal
   version          40)

(object Design "Logical View"
   is_unit          TRUE
   is_loaded        TRUE
   file_name        "$DATA\\demo1.mdl"
   quid             "3353F13A0384"
   defaults         (object defaults
                    rightMargin       0.250000
                    leftMargin        0.250000
                    topMargin         0.250000
                    bottomMargin      0.500000
                    pageOverlap       0.250000
                    clipIconLabels    TRUE
                    autoResize        TRUE
                    snapToGrid        TRUE
                    gridX             16
                    gridY             16
                    defaultFont       (object Font
                       size           9
                       face           "helvetica"
                       bold           FALSE
                       italics        FALSE
                       underline      FALSE
                       strike         FALSE
                       color          0
                       default_color  TRUE)
                    showMessageNum    1
                    showClassOfObject TRUE
                    notation          "Unified")
   root_usecase_package               (object Class_Category "Use Case View"
                    quid              "3353F13A0386"
                    exportControl     "Public"
                    global            TRUE
                    logical_models    (list unit_reference_list
                       (object Class "Student"
                                      quid             "3353F162000A"
                                      documentation    "Someone who is registered to take classes at the University."
                                      stereotype       "Actor")

                 ......
```

FIGURE 39

Code	Parent	Number	Description	Keyword	Comment	Category	Verification	Leaf
Ex-a	0	1	Define element	-	-	-	-	FALSE
EX-a	0	2	Specify drawing environment	-	-	-	-	FALSE
EX-a	1	1	Define line element	-	-	-	-	FALSE
EX-a	1	2	Define rectangle element	-	-	-	-	FALSE
EX-a	1	3	Define circle element	-	-	-	-	FALSE
EX-a	1.1	1	Define start	-	-	-	-	TRUE
EX-a	1.1	2	Define end	-	-	-	-	TRUE
EX-a	1.2	1	Define upper left corner	-	-	-	-	TRUE
EX-a	1.2	2	Define lower right corner	-	-	-	-	TRUE
EX-a	1.3	1	Define center	-	-	-	-	TRUE
EX-a	1.3	2	Define radius	-	-	-	-	TRUE
EX-a	2	1	Identify the drawing type	-	-	-	-	FALSE
EX-a	2	2	Detect drawing location	-	-	-	-	FALSE
EX-a	2	3	Draw the element	-	-	-	-	TRUE
EX-a	2.1	1	Identify line	-	-	-	-	TRUE
EX-a	2.1	2	Identify rectangle	-	-	-	-	TRUE
EX-a	2.1	3	Identify circle	-	-	-	-	TRUE
EX-a	2.2	1	Detect mouse push	-	-	-	-	TRUE
EX-a	2.2	2	Detect mouse release	-	-	-	-	TRUE

FIGURE 40

Code	Parent	Number	Alternative	Description	Keyword	Comment	Category	Verification	Leaf
Ex-a	0	1	0	Element characteristics	-	-	-	-	FALSE
EX-a	0	2	0	GUI with window	-	-	-	-	FALSE
EX-a	1	1	0	Line characteristics	-	-	-	-	FALSE
EX-a	1	2	0	Rectangle characteristics	-	-	-	-	FALSE
EX-a	1	3	0	Circle characteristics	-	-	-	-	FALSE
EX-a	1.1	1	0	Start point	-	-	-	-	TRUE
EX-a	1.1	2	0	End point	-	-	-	-	TRUE
EX-a	1.2	1	0	Upper left point	-	-	-	-	TRUE
EX-a	1.2	2	0	Lower right point	-	-	-	-	TRUE
EX-a	1.3	1	0	Center point	-	-	-	-	TRUE
EX-a	1.3	2	0	Radius	-	-	-	-	TRUE
EX-a	2	1	0	Radio buttons	-	-	-	-	FALSE
EX-a	2	2	0	Mouse click information	-	-	-	-	FALSE
EX-a	2	3	0	Drawign area	-	-	-	-	TRUE
EX-a	2.1	1	0	Line button	-	-	-	-	TRUE
EX-a	2.1	2	0	Rectangle button	-	-	-	-	TRUE
EX-a	2.1	3	0	Circle button	-	-	-	-	TRUE
EX-a	2.2	1	0	Event for push	-	-	-	-	TRUE
EX-a	2.2	2	0	Event for release	-	-	-	-	TRUE

Name	Attribute	Type	Description
Line_d	start	Point	-
Line_d	end	Point	-
Rectangle_d	upper_left	Point	-
Rectangle_d	lower_right	Point	-
Circle_d	center	Point	-
Circle_d	radius	Radius	-
Element_d	line	Line_d	-
Element_d	rectangle	Rectangle_d	-
Element_d	circle	Circle_d	-
Window_d	line	Radiobutton	-
Window_d	rectangle	Radiobutton	-
Window_d	circle	Radiobutton	-
Window_d	mouse	Mouse	-
Window_d	canvas	Canvas	-

DP Table

ATTRIBUTE Table

VARIABLE Table

FIGURE 41

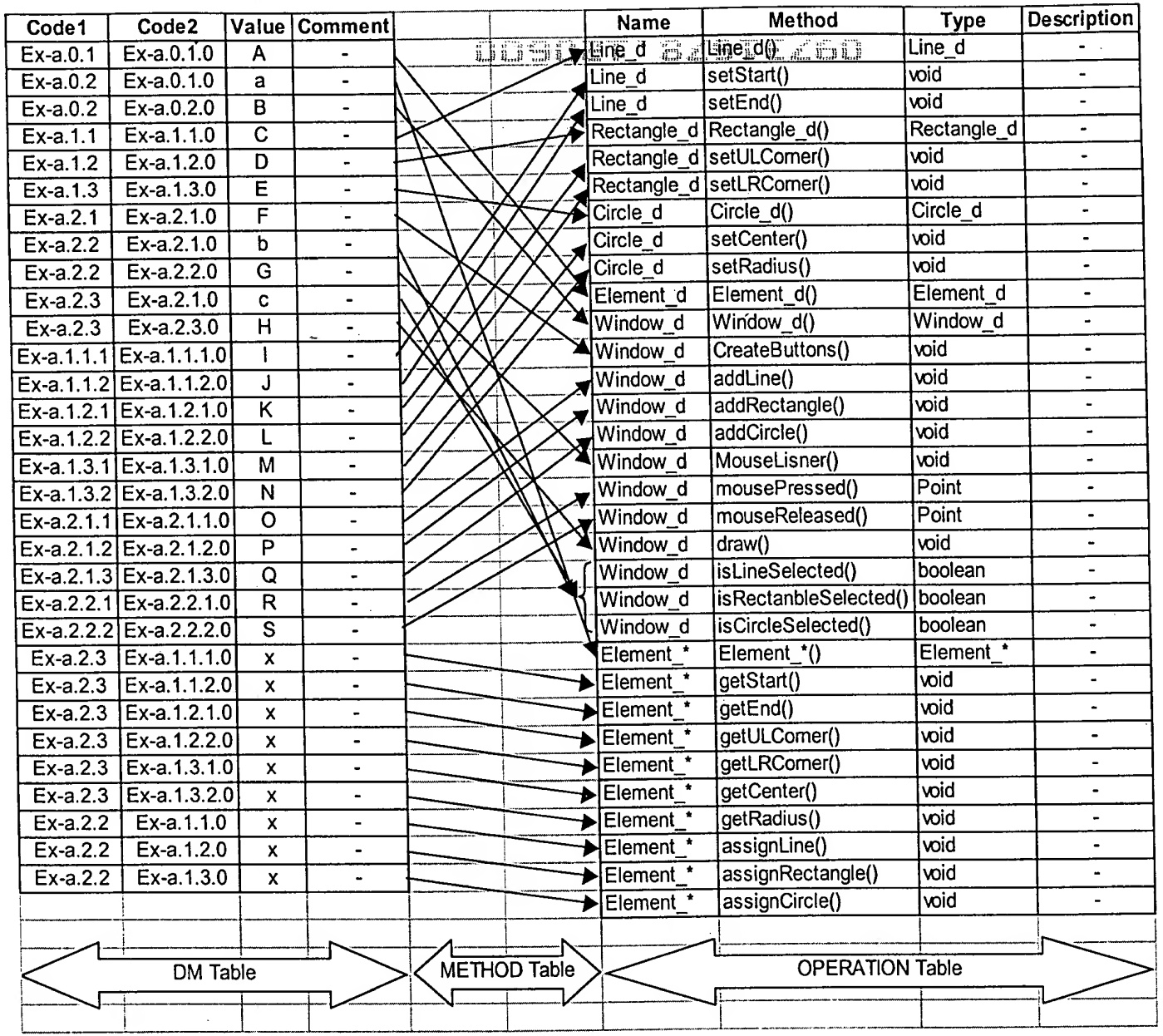

Code1
Code2
Value
Comment
Ex-a.0.1 Ex-a.0.1.0 A -
Ex-a.0.2 Ex-a.0.1.0 a -
Ex-a.0.2 Ex-a.0.2.0 B -
Ex-a.1.1 Ex-a.1.1.0 C -
Ex-a.1.2 Ex-a.1.2.0 D -
Ex-a.1.3 Ex-a.1.3.0 E -
Ex-a.2.1 Ex-a.2.1.0 F -
Ex-a.2.2 Ex-a.2.1.0 b -
Ex-a.2.2 Ex-a.2.2.0 G -
Ex-a.2.3 Ex-a.2.1.0 c -
Ex-a.2.3 Ex-a.2.3.0 H -
Ex-a.1.1.1 Ex-a.1.1.1.0 I -
Ex-a.1.1.2 Ex-a.1.1.2.0 J -
Ex-a.1.2.1 Ex-a.1.2.1.0 K -
Ex-a.1.2.2 Ex-a.1.2.2.0 L -
Ex-a.1.3.1 Ex-a.1.3.1.0 M -
Ex-a.1.3.2 Ex-a.1.3.2.0 N -
Ex-a.2.1.1 Ex-a.2.1.1.0 O -
Ex-a.2.1.2 Ex-a.2.1.2.0 P -
Ex-a.2.1.3 Ex-a.2.1.3.0 Q -
Ex-a.2.2.1 Ex-a.2.2.1.0 R -
Ex-a.2.2.2 Ex-a.2.2.2.0 S -
Ex-a.2.3 Ex-a.1.1.1.0 x -
Ex-a.2.3 Ex-a.1.1.2.0 x -
Ex-a.2.3 Ex-a.1.2.1.0 x -
Ex-a.2.3 Ex-a.1.2.2.0 x -
Ex-a.2.3 Ex-a.1.3.1.0 x -
Ex-a.2.3 Ex-a.1.3.2.0 x -
Ex-a.2.2 Ex-a.1.1.0 x -
Ex-a.2.2 Ex-a.1.2.0 x -
Ex-a.2.2 Ex-a.1.3.0 x -
Name
Method
Type
Description
Line_d Line_d() Line_d -
Line_d setStart() void -
Line_d setEnd() void -
Rectangle_d Rectangle_d() Rectangle_d -
Rectangle_d setULCorner() void -
Rectangle_d setLRCorner() void -
Circle_d Circle_d() Circle_d -
Circle_d setCenter() void -
Circle_d setRadius() void -
Element_d Element_d() Element_d -
Window_d Window_d() Window_d -
Window_d CreateButtons() void -
Window_d addLine() void -
Window_d addRectangle() void -
Window_d addCircle() void -
Window_d MouseLisner() void -
Window_d mousePressed() Point -
Window_d mouseReleased() Point -
Window_d draw() void -
Window_d isLineSelected() boolean -
Window_d isRectanbleSelected() boolean -
Window_d isCircleSelected() boolean -
Element_* Element_*() Element_* -
Element_* getStart() void -
Element_* getEnd() void -
Element_* getULCorner() void -
Element_* getLRCorner() void -
Element_* getCenter() void -
Element_* getRadius() void -
Element_* assignLine() void -
Element_* assignRectangle() void -
Element_* assignCircle() void -
DM Table
METHOD Table
OPERATION Table

FIGURE 42

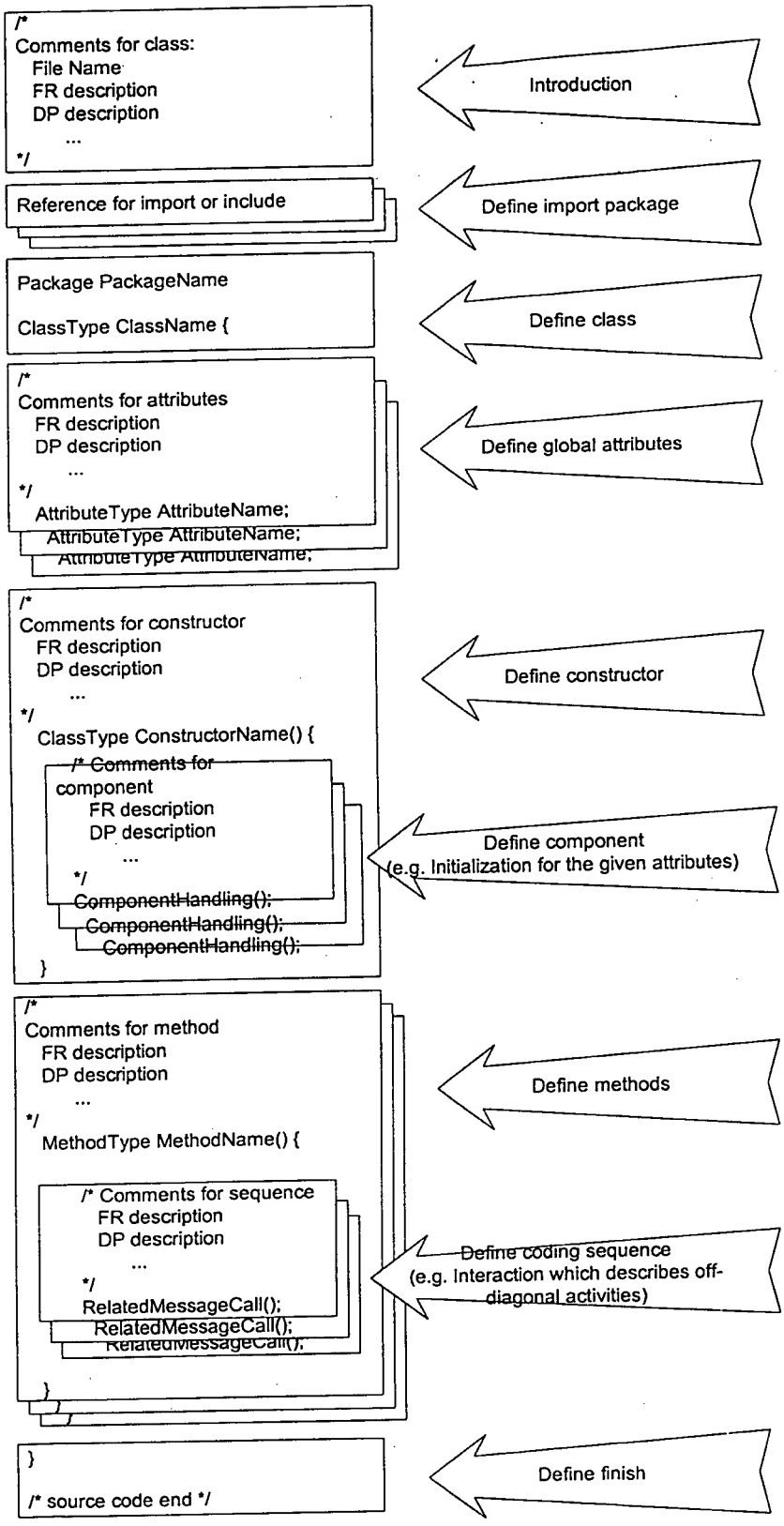

/*
Comments for class:
File Name
FR description
DP description
...
*/
Introduction
Reference for import or include
Define import package
Package PackageName
ClassType ClassName {
Define class
/*
Comments for attributes
FR description
DP description
...
*/
AttributeType AttributeName;
AttributeType AttributeName;
AttributeType AttributeName;
Define global attributes
/*
Comments for constructor
FR description
DP description
...
*/
ClassType ConstructorName() {
/* Comments for
component
FR description
DP description
...
*/
ComponentHandling();
ComponentHandling();
ComponentHandling();
}
Define constructor
Define component
(e.g. Initialization for the given attributes)
/*
Comments for method
FR description
DP description
...
*/
MethodType MethodName() {
/* Comments for sequence
FR description
DP description
...
*/
RelatedMessageCall();
RelatedMessageCall();
RelatedMessageCall();
}
Define methods
Define coding sequence
(e.g. Interaction which describes off-diagonal activities)
}
/* source code end */
Define finish

FIGURE 43

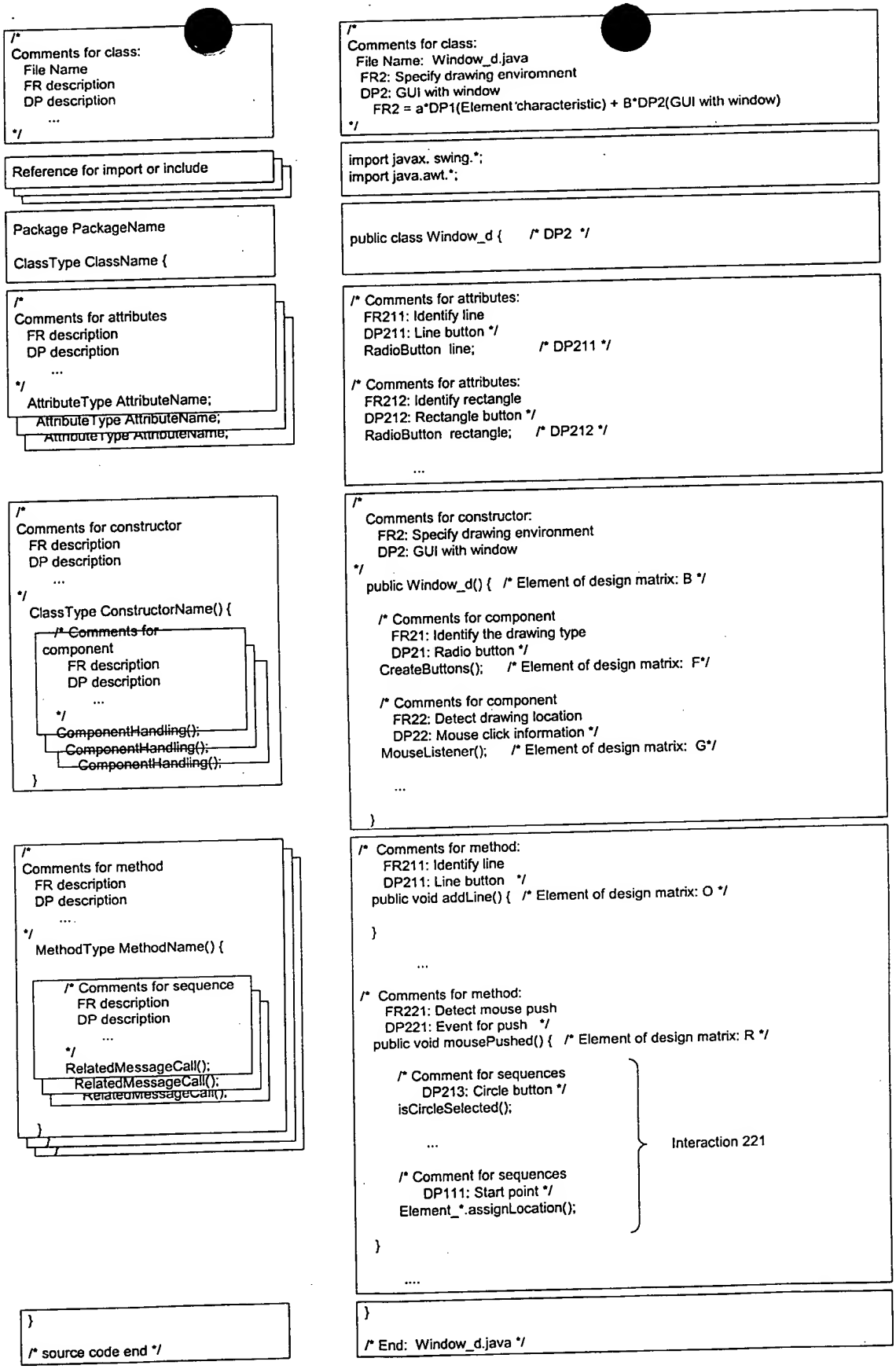

/*
Comments for class:
File Name
FR description
DP description
...
*/
Reference for import or include
Package PackageName
ClassType ClassName {
/*
Comments for attributes
FR description
DP description
...
*/
AttributeType AttributeName;
/*
Comments for constructor
FR description
DP description
...
*/
ClassType ConstructorName() {
/* Comments for
component
FR description
DP description
...
*/
ComponentHandling();
}
/*
Comments for method
FR description
DP description
...
*/
MethodType MethodName() {
/* Comments for sequence
FR description
DP description
...
*/
RelatedMessageCall();
}
/* source code end */
/*
Comments for class:
File Name: Window_d.java
FR2: Specify drawing enviromnent
DP2: GUI with window
FR2 = a*DP1(Element characteristic) + B*DP2(GUI with window)
*/
import javax. swing.*;
import java.awt.*;
public class Window_d { /* DP2 */
/* Comments for attributes:
FR211: Identify line
DP211: Line button */
RadioButton line; /* DP211 */
/* Comments for attributes:
FR212: Identify rectangle
DP212: Rectangle button */
RadioButton rectangle; /* DP212 */
...
/*
Comments for constructor:
FR2: Specify drawing environment
DP2: GUI with window
*/
public Window_d() { /* Element of design matrix: B */
/* Comments for component
FR21: Identify the drawing type
DP21: Radio button */
CreateButtons(); /* Element of design matrix: F*/
/* Comments for component
FR22: Detect drawing location
DP22: Mouse click information */
MouseListener(); /* Element of design matrix: G*/
...
}
/* Comments for method:
FR211: Identify line
DP211: Line button */
public void addLine() { /* Element of design matrix: O */
}
...
/* Comments for method:
FR221: Detect mouse push
DP221: Event for push */
public void mousePushed() { /* Element of design matrix: R */
/* Comment for sequences
DP213: Circle button */
isCircleSelected();
...
/* Comment for sequences
DP111: Start point */
Element_*.assignLocation();
Interaction 221
}
....
}
/* End: Window_d.java */

FIGURE 44

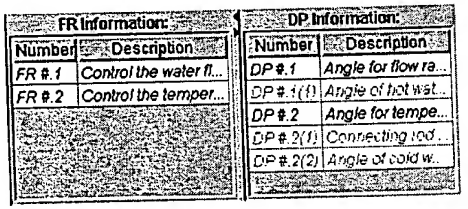
FR Information:	
Number	Description
FR #.1	Provide security
FR #.2	Assign tasks
FR #.3	Manage schedule
FR #.4	Construct design h...
FR #.5	Facilitate changes ...

DP Information:	
Num...	Description
DP #.1	Login privilege
DP #.2	Resource of desig...
DP #.3	Schedule-managi...
DP #.4	Data structure for ...
DP #.5	ECO handling tool

FIGURE 45A

	FR	DP
1	FR 1 description ↔	DP 1 description
2	FR 2 description ↔	DP 2 description
3	FR 3 description ↔	DP 3 description

FIGURE 45B



FR Information:	
Number	Description
FR #.1	Control the water fl...
FR #.2	Control the temper...

DP Information:	
Number	Description
DP #.1	Angle for flow ra...
DP #.1(1)	Angle of hot wat...
DP #.2	Angle for tempe...
DP #.2(1)	Connecting rod ...
DP #.2(2)	Angle of cold w...

FIGURE 46A

	FR	DP
1	FR 1 description	DP 1 description
2	FR 2 description	Alternative DP 2(a)
		Alternative DP 2(b)
		Alternative DP 2(c)
3	FR 3 description	DP 3 description

FIGURE 46B

Parent Information:

Number	Description
FR 1.1	Manage design workflow
DP 1.1	Management roadmap

FR Information:

Number	Description
FR #.1	Provide security
FR #.2	Assign tasks
FR #.3	Manage schedule
FR #.4	Construct design h...
FR #.5	Facilitate changes ...

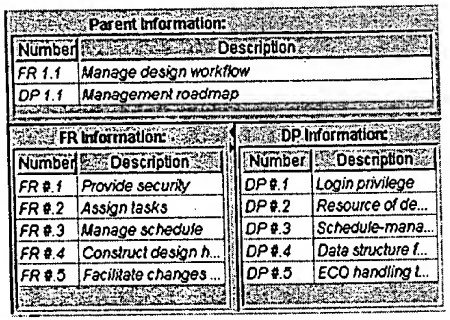
DP Information:

Number	Description
DP #.1	Login privilege
DP #.2	Resource of de...
DP #.3	Schedule-mana...
DP #.4	Data structure f...
DP #.5	ECO handling f...

FIGURE 47A

	FR	DP
Parent	Parent FR description	Parent DP description
1	FR 1 description	DP 1 description
2	FR 2 description	Alternative DP 2(a)
		Alternative DP 2(b)
		Alternative DP 2(c)
3	FR 3 description	DP 3 description

FIGURE 47B



Parent Information:	
Number	Description
FR 1.1	Manage design workflow
DP 1.1	Management roadmap

FR Information:	
Number	Description
FR #.1	Provide security
FR #.2	Assign tasks
FR #.3	Manage schedule
FR #.4	Construct design h...
FR #.5	Facilitate changes ...

DP Information:	
Number	Description
DP #.1	Login privilege
DP #.2	Resource of de...
DP #.3	Schedule-mana...
DP #.4	Data structure f...
DP #.5	ECO handling t...

FIGURE 48A

#: 1.2.3	FR	DP
Parent	Parent FR description	Parent DP description
#.1	FR 1 description	DP 1 description
#.2	FR 2 description	Alternative DP 2(a)
		Alternative DP 2(b)
		Alternative DP 2(c)
#.3	FR 3 description	DP 3 description

FIGURE 48A/B

Constraint Information:

Num...	Descr...	FR #.1	FR #.2	FR #.3	FR #.4	FR #.5
C #.1	Make...	X	X	X	X	X
C #.2	Supp...	X	X	X	X	X
C #.3	Elimi...	X	X	X	X	X
C #.4	Facilit...	X	X	X	X	X
C #.5	Funct...			X	X	
C #.6	Obje...			X	X	

FIGURE 49A

Mapping | Constraints | Robust Design | Analysis

Index	Information				Target Value		
	Category	Type	Constraints	Comments	Operator	Target	Calculated
1	Critical	Marke	Weight	☑	Less than (<+)	300 lb	
2	Interface	Field	Cost	☐	More than (>=)	$500	
3	Project	Manu	Volume	☑	Exact (= +/-)	10cu	

	C#.1	C#.2	C#.3	CA's
FR#.1	X			☑
FR#.2	X	X	X	☐
FR#.3	X		X	☐
FR#.4	X			☑

FIGURE 49B

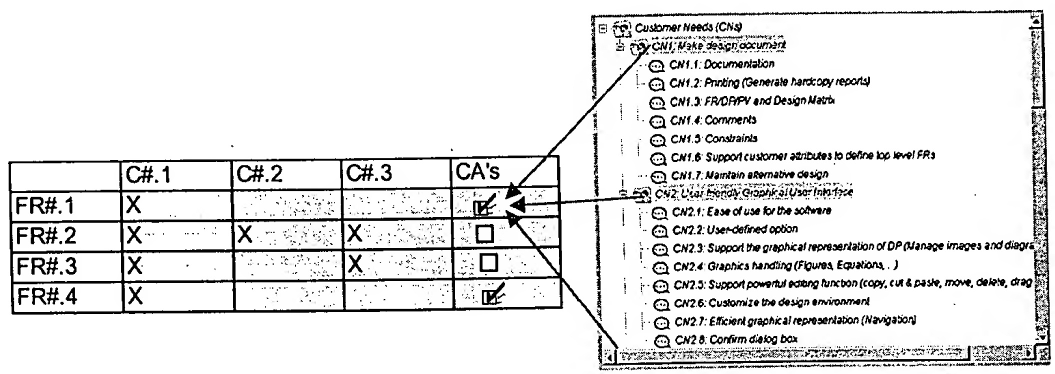

C#.1
C#.2
C#.3
CA's
FR#.1
FR#.2
FR#.3
FR#.4
X
Customer Needs (CNs)
CN1: Make design document
CN1.1: Documentation
CN1.2: Printing (Generate hardcopy reports)
CN1.3: FR/DP/PV and Design Matrix
CN1.4: Comments
CN1.5: Constraints
CN1.6: Support customer attributes to define top level FRs
CN1.7: Maintain alternative design
CN2: User friendly Graphical User Interface
CN2.1: Ease of use for the software
CN2.2: User-defined option
CN2.3: Support the graphical representation of DP (Manage images and diagra
CN2.4: Graphics handling (Figures, Equations, ..)
CN2.5: Support powerful editing function (copy, cut & paste, move, delete, drag
CN2.6: Customize the design environment
CN2.7: Efficient graphical representation (Navigation)
CN2.8: Confirm dialog box

FIGURE 50

Index	Information				Target Value		
#	Category	Type	Constraints	Comments	Operator	Target	Calculated
1	Critical	Marke	Weight	☑	Less than (<+)	300 lb	
2	Interface	Field	Cost	☐	More than (>=)	$500	
3	Project	Manu	Volume	☑	Exact (= +/-)	10cu	

FIGURE 51

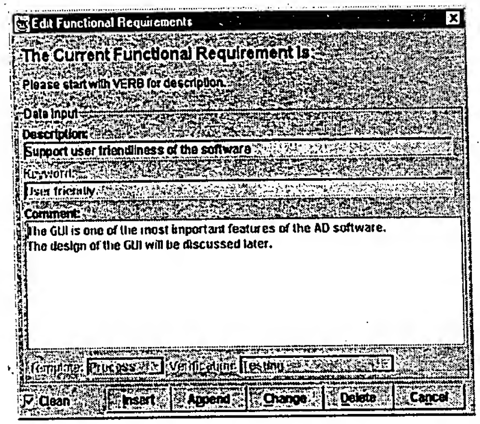


FIGURE 52A

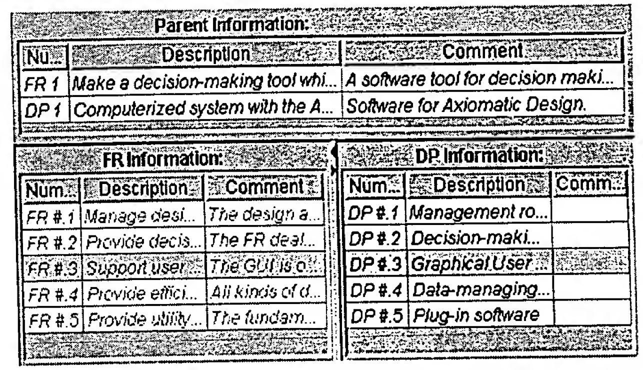


FIGURE 52B

Index	Information			Comment		App. Link
#	Template	FR	DP	FR	DP	
Parent		Control the FR/DP domain	FR/DP window	☑	☐	
1		Control the mapping	Mapping tab	☑	☑	
			Domain tab	☐	☑	
2		Assign constraints	Constraints tab	☐	☐	
3		Refine the design	Robust design tab	☑	☐	
4		Analyze the design	Analysis tab	☑	☑	

FIGURE 52C

			DP1						DP2					
			DP11		DP12		DP13		DP21			DP22		
			DP111	DP112	DP121	DP122	DP131	DP132	DP211	DP212	DP213	DP221	DP222	DP23
FR1	FR11	FR111	X											
		FR112		X										
	FR12	FR121			X									
		FR122				X								
	FR13	FR131					X							
		FR132						X						
FR2	FR21	FR211							X					
		FR212								X				
		FR213									X			
	FR22	FR221	X		X		X		X	X	X	X		
		FR222		X		X		X	X	X	X		X	
	FR23		X	X	X	X	X	X	X	X	X			X

FIGURE 53

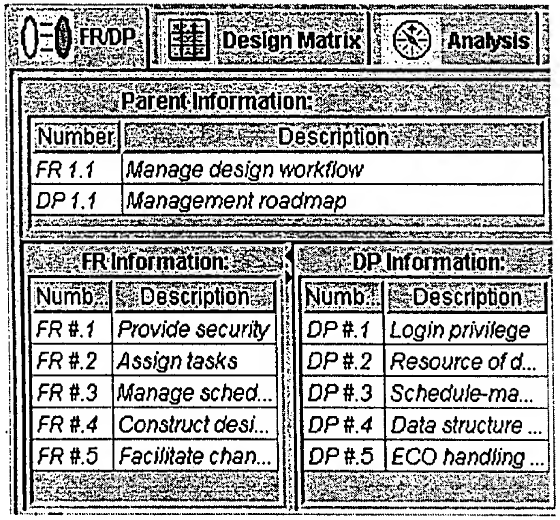


FIGURE 54A

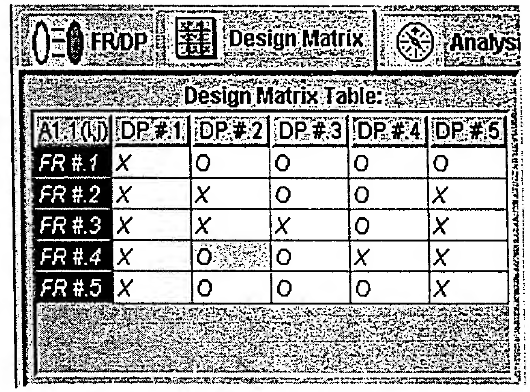


FIGURE 54B

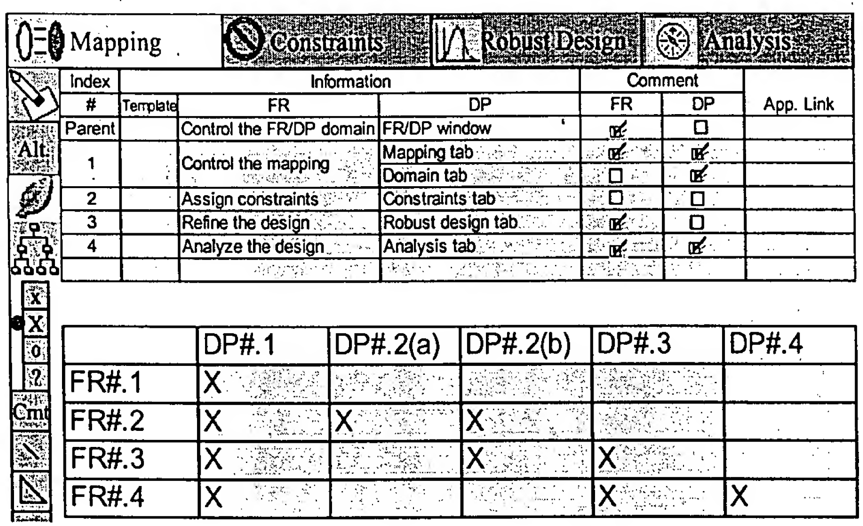


FIGURE 54C

FIGURE 55A

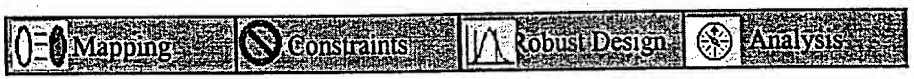


FIGURE 55B

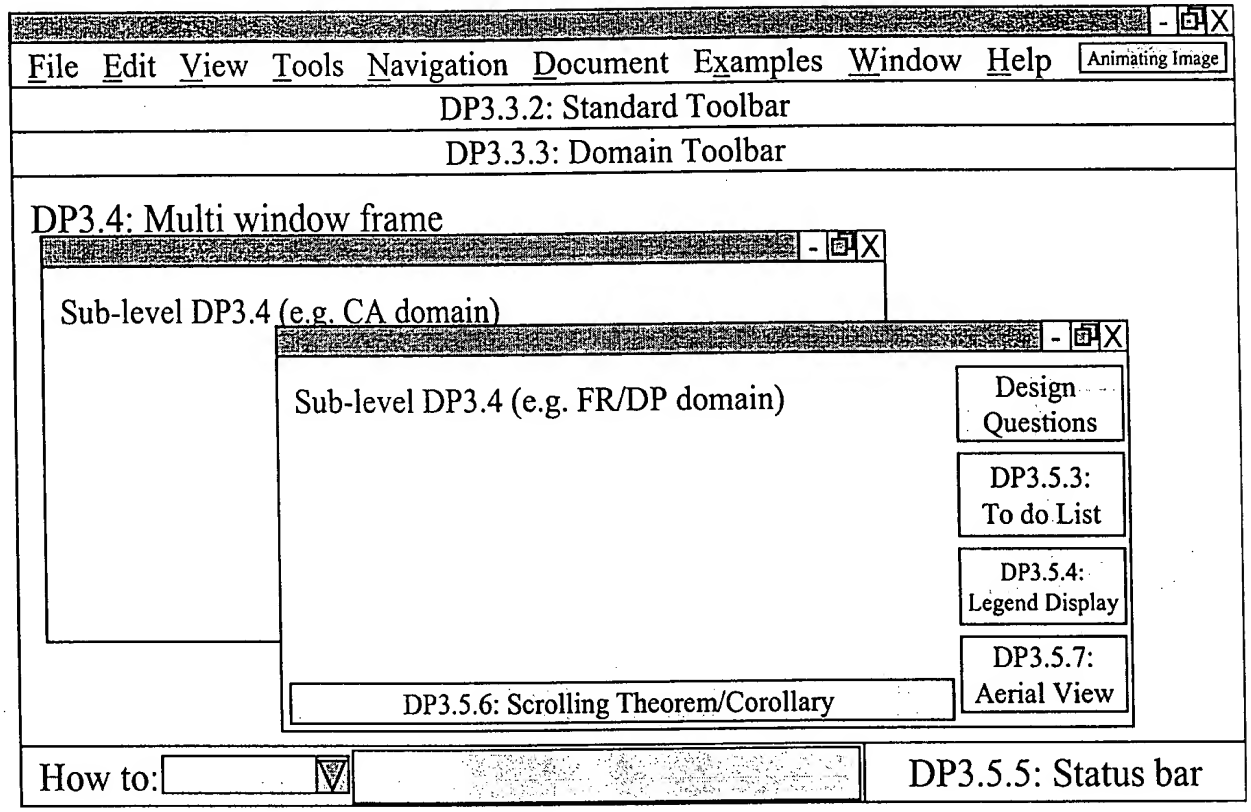


FIGURE 56

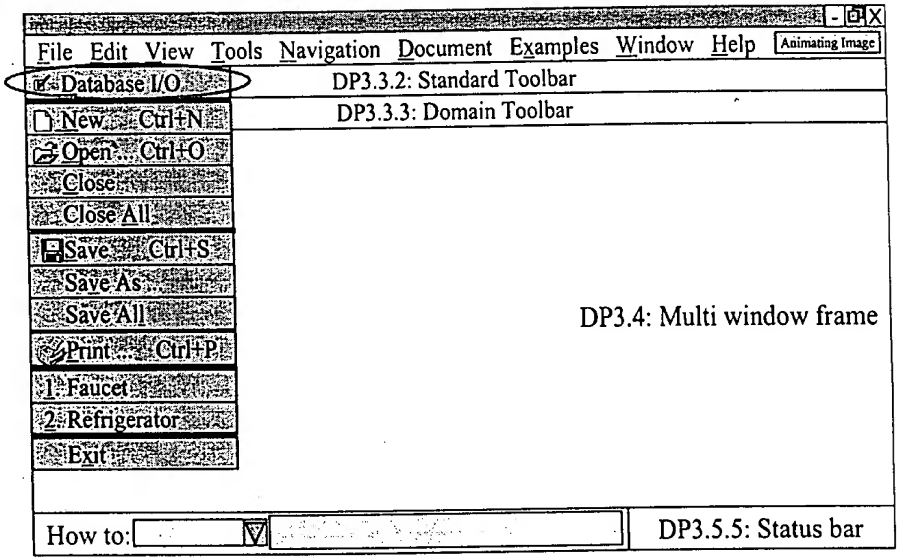


FIGURE 57

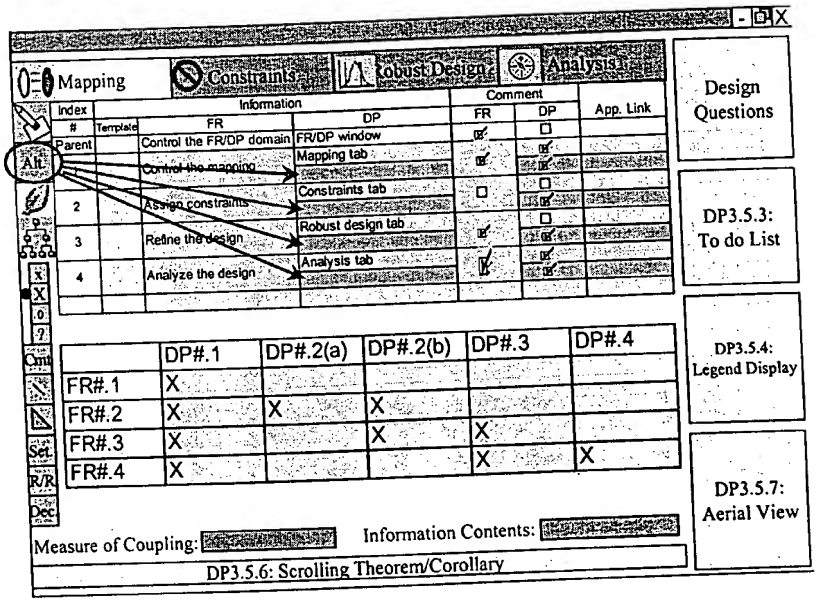


FIGURE 58

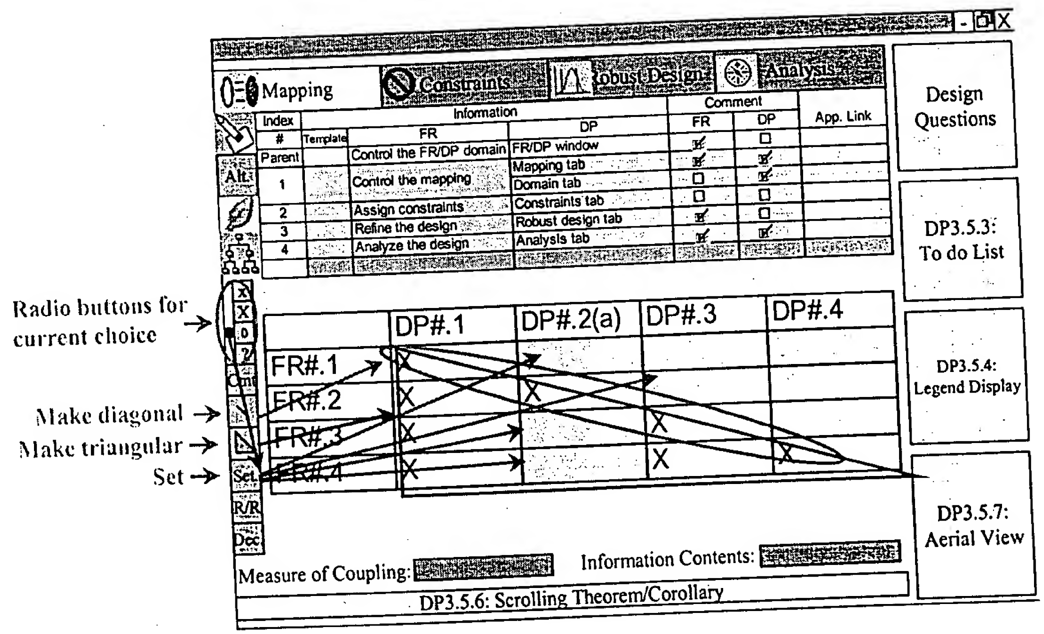


FIGURE 59

Roadmap		Is this step finished?		Resources for control						
				Menu	Tab	Toolbar	Buttons			
		Yes	No				In Mapping tap	In Constraint tab	In Analysis tab	In Robust Design
Start the design process		Disable		*View -> Project Control*	*Constraints*, Robust design, Analysis	Project Control				
Activities at one level of the design hierarchy	FR/DP mapping	Enable			*Constraints*		*One step design matrix control buttons*			
			Disable				Decompose			
	Define Design Matrix	Enable			Analysis		Decompose		Flow Chart, Impact List, Check consistency	
			Disable						Flow Chart, Impact List, Check consistency	
Activities over the design hierarchy	Define leaf level	Enable		*View -> Project Control*	Robust design	Project Control			*Check Constraints, Audit*	
			Disable						*Check Constraints, Audit*	

FIGURE 60

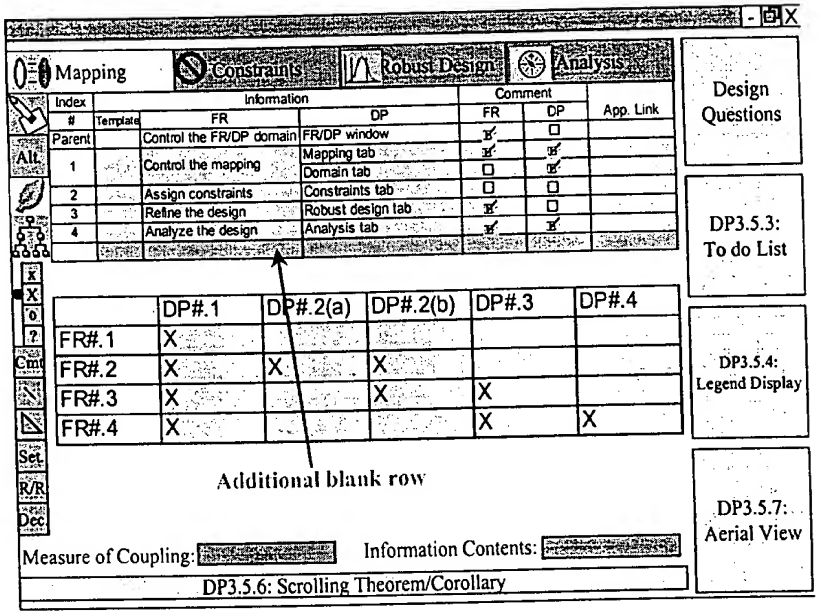


FIGURE 61

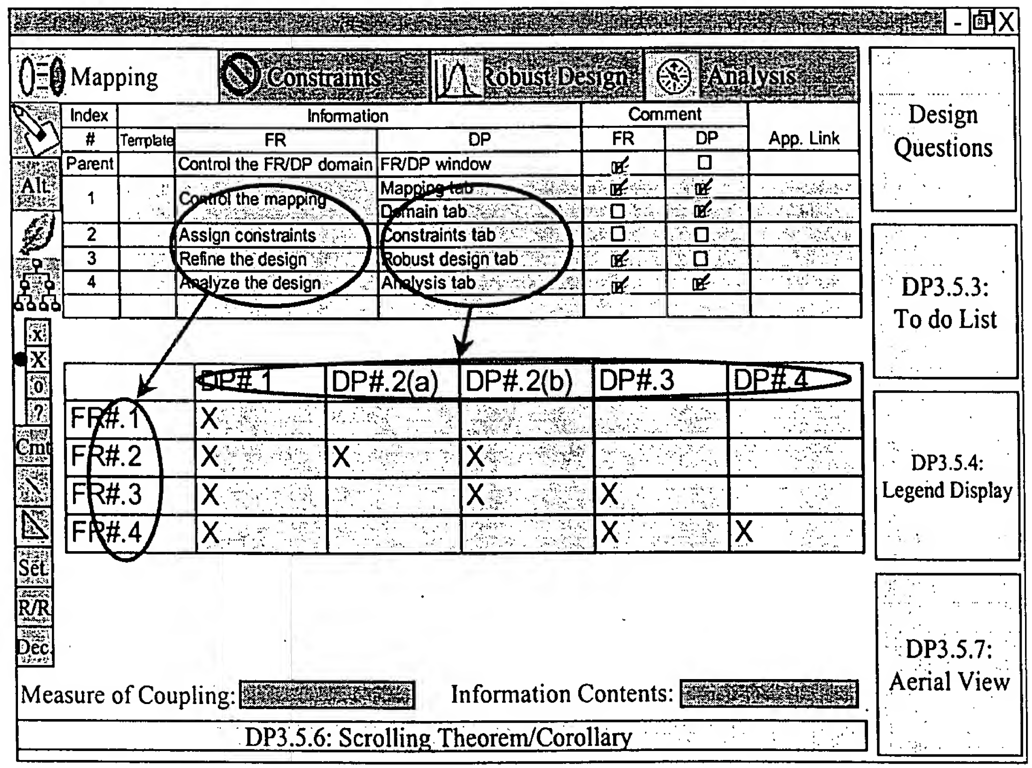


FIGURE 62

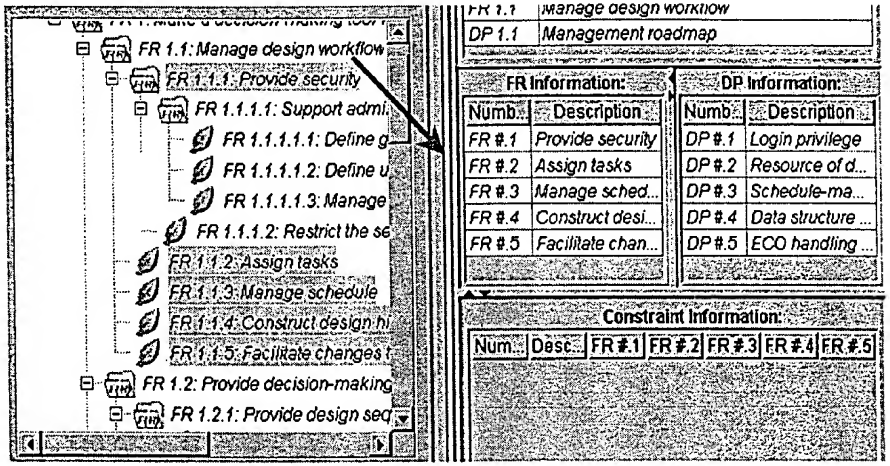


FIGURE 63A

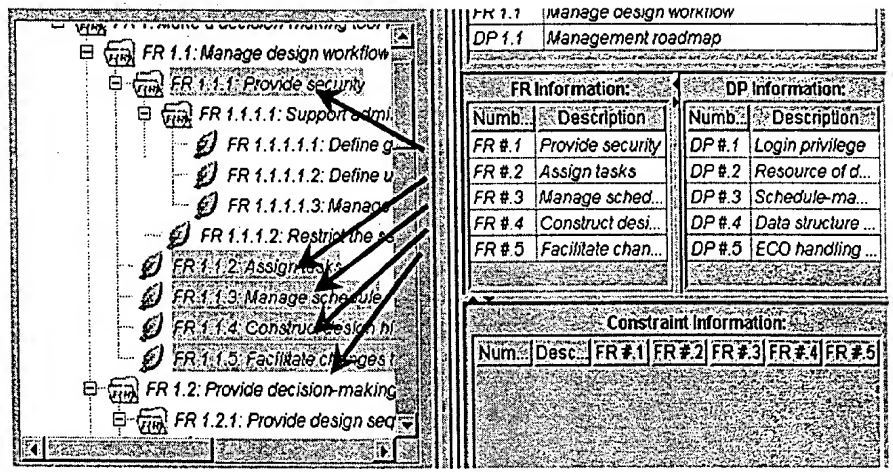


FIGURE 63B

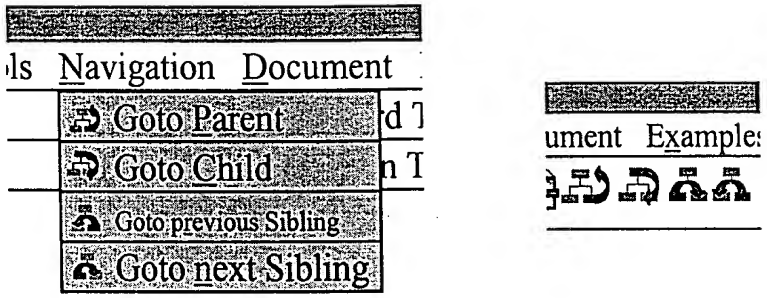

ls Navigation Document
Goto Parent
Goto Child
Goto previous Sibling
Goto next Sibling
ument Examples

FIGURE 64

Control Item			Level 1 Beginner	Level 2	Level 3 Intermediate	Level 4	Level 5 Expert
Available Features		FR/DP Mapping	●	●	●	●	●
		Design Matrix	●	●	●	●	●
		Alternativ DP		●	●	●	●
		Analysis-Flow Chart		●	●	●	●
		Constraints			●	●	●
		Comments			●	●	●
		CN			●	●	●
		CN/FR Mapping			●	●	●
		Analysis-Child List			●	●	●
		Analysis-Impact List			●	●	●
		DP/PV Mapping				●	●
		Analysis-Check Consistency				●	●
		Analysis-Check Constraints				●	●
		Templates				●	●
		Verification				●	●
		Application Link				●	●
		Analysis-Audit					●
		Nested(Full) Matrix Handling					●
		Robust Design					●
		Project Control					●
Automatic Menu Control (Enables the marked item)	File Menu	Database I/O				◉	◉
	View Menu	CN Domain			◉	◉	◉
		FR/DP Domain	◉	◉	◉	◉	◉
		DP/PV Domain				◉	◉
		Nested (Full) Matrix					◉
		Project Control					◉
	Preference Menu	Display Configuration Manag	◉	◉	◉	◉	◉
		Numbering		◉	◉	◉	◉
		Design Matrix		◉	◉	◉	◉
		Display Color		◉	◉	◉	◉
		Design Matrix Color	◉	◉	◉	◉	◉
		GUI Display			◉	◉	◉
		File Location				◉	◉
		Resource				◉	◉
		Database I/O				◉	◉
		Templates				◉	◉
		Constraints				◉	◉
		Verifications				◉	◉
	Document Menu	PV Tree Diagram				◉	◉
		Nested(Full) Matrix					◉
Automatic Window Control (Displays the marked item)	FR/DP Window	No Tab	◉				
		Mapping Tab		◉	◉	◉	◉
		Constraints Tab			◉	◉	◉
		Robust Design Tab					◉
		Analysis Tab: Flow Chart Tab		◉	◉	◉	◉
		Analysis Tab: Child List Tab			◉	◉	◉
		Analysis Tab: Impact List Tab			◉	◉	◉
		Analysis Tab: Check Consistency Tab				◉	◉
		Analysis Tab: Check Constraints Tab				◉	◉
		Analysis Tab: Audit Tab					◉
	CN Window				◉	◉	◉
	DP/PV Window					◉	◉
	Project Control Window						◉
	Nested (Full) Design Matrix Window						◉

FIGURE 65

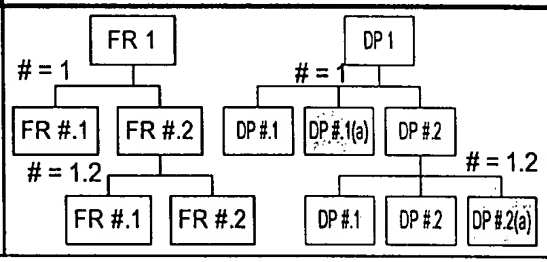
		Default Numbering	Alternative Numbering	Example
Numbering Type	Numeric	☑	☐	1, 2, 3
	Lower case	☐	☑	a, b, c
	Upper case	☐	☐	A, B, C
Indicator	Alternative connector	()		Defined by user
	Parent index	#		
	Divider	.		
Example				

FIGURE 66

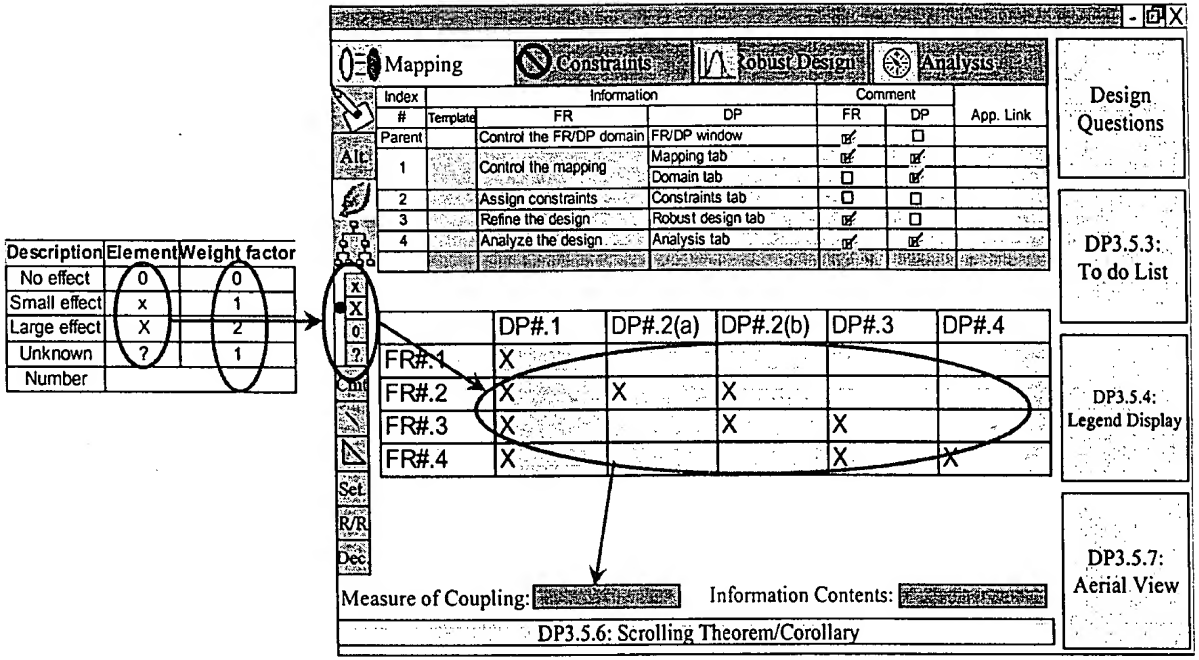


FIGURE 67

		Legend category		
		Color	Font	Line
Display	Activited cell			N/A
	Normal			
	Default			N/A
	Focus			N/A
	Alternative			N/A
	Redundant			N/A
	Constraints			N/A
	Comments			N/A
Design Matrix	Uncoupled		N/A	
	Decoupled		N/A	
	Coupled		N/A	
	Undefined		N/A	
Template	Process			—
	Transport			---
	...			

FIGURE 68

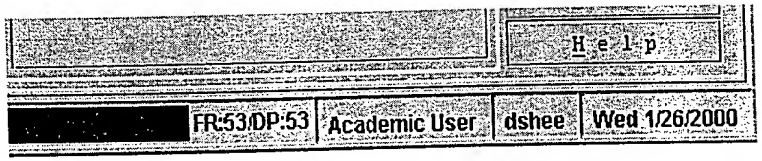


FIGURE 69

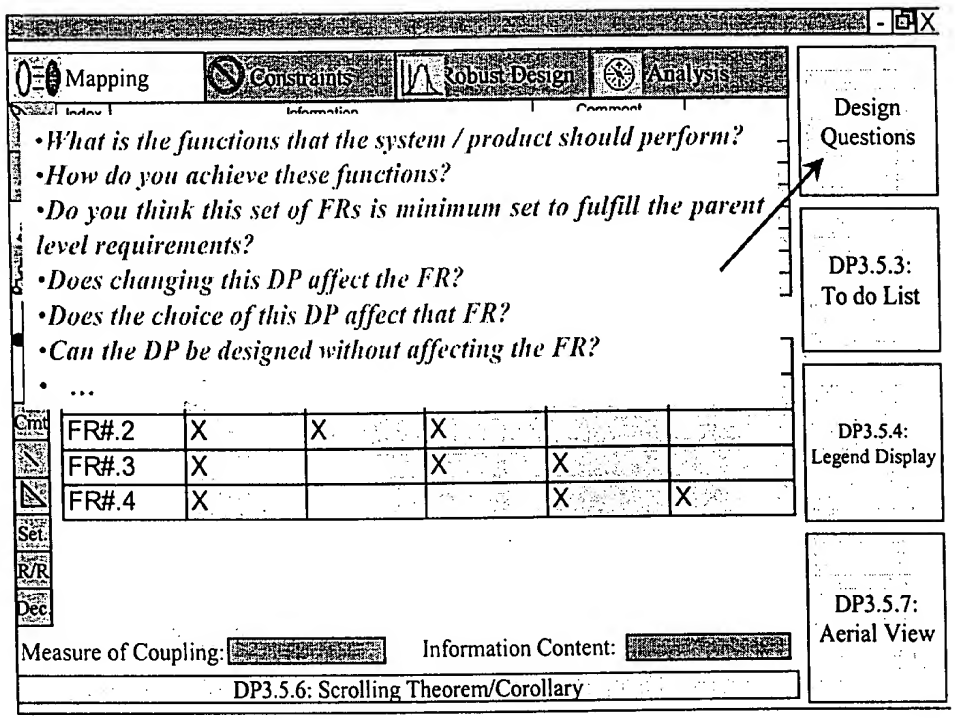

Mapping
Constraints
Robust Design
Analysis
•What is the functions that the system / product should perform?
•How do you achieve these functions?
•Do you think this set of FRs is minimum set to fulfill the parent level requirements?
•Does changing this DP affect the FR?
•Does the choice of this DP affect that FR?
•Can the DP be designed without affecting the FR?
• ...
FR#.2 X X X
FR#.3 X X X
FR#.4 X X X
Cmt
Set
R/R
Dec
Measure of Coupling:
Information Content:
DP3.5.6: Scrolling Theorem/Corollary
Design Questions
DP3.5.3: To do List
DP3.5.4: Legend Display
DP3.5.7: Aerial View

FIGURE 70

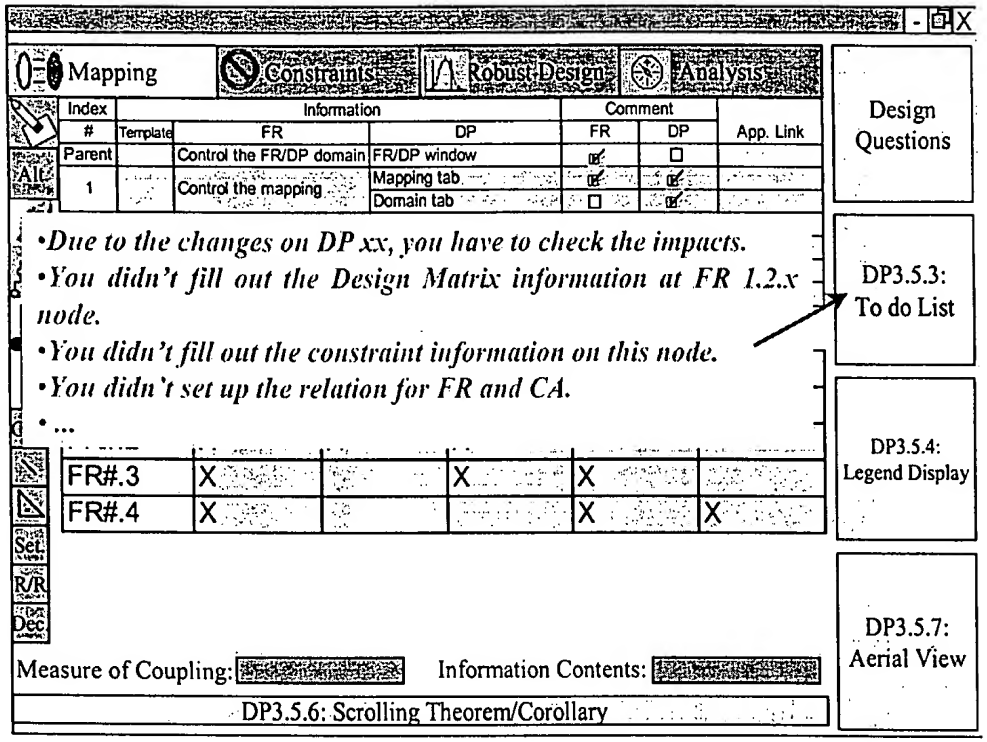

Mapping
Constraints
Robust Design
Analysis
Index
#
Template
Information
FR
DP
Comment
FR
DP
App. Link
Parent
Control the FR/DP domain
FR/DP window
1
Control the mapping
Mapping tab
Domain tab
•Due to the changes on DP xx, you have to check the impacts.
•You didn't fill out the Design Matrix information at FR 1.2.x node.
•You didn't fill out the constraint information on this node.
•You didn't set up the relation for FR and CA.
•...
FR#.3
FR#.4
Measure of Coupling:
Information Contents:
DP3.5.6: Scrolling Theorem/Corollary
Design Questions
DP3.5.3: To do List
DP3.5.4: Legend Display
DP3.5.7: Aerial View

FIGURE 71

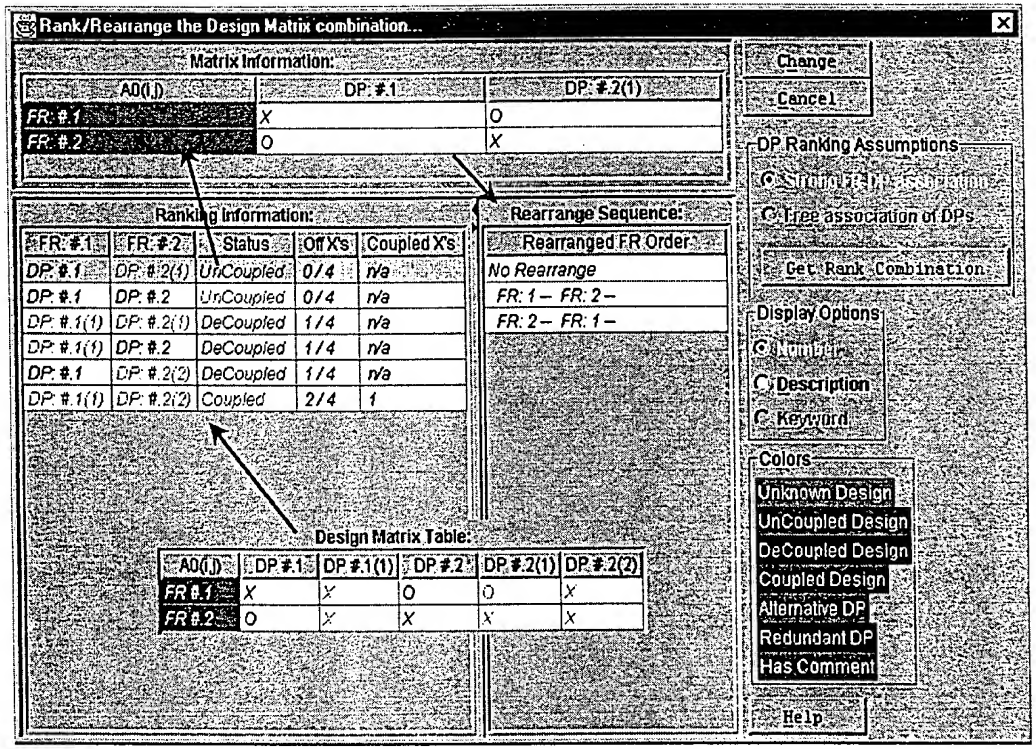


FIGURE 72

Child List | Impact List | Inconsistency | Decoupling

Number	FR Description	DP Description
1.1	Manage design workflow	Management roadmap
1.1.1	Provide security	Login privilege
1.1.2	Assign tasks	Resource of design activity
1.1.3	Manage schedule	Schedule-managing tool (e.g. MS Project)
1.1.4	Construct design hierarchy	Data structure for Axiomatic Design concept
1.1.5	Facilitate changes to the design	ECO handling tool
1.1.1.1	Support administrative tool	User manager
1.1.1.2	Restrict the security access level	Authority code
1.1.1.1.1	Define group	Group specification
1.1.1.1.2	Define user	User specification
1.1.1.1.3	Manage authority code	Authority code specification

FIGURE 73

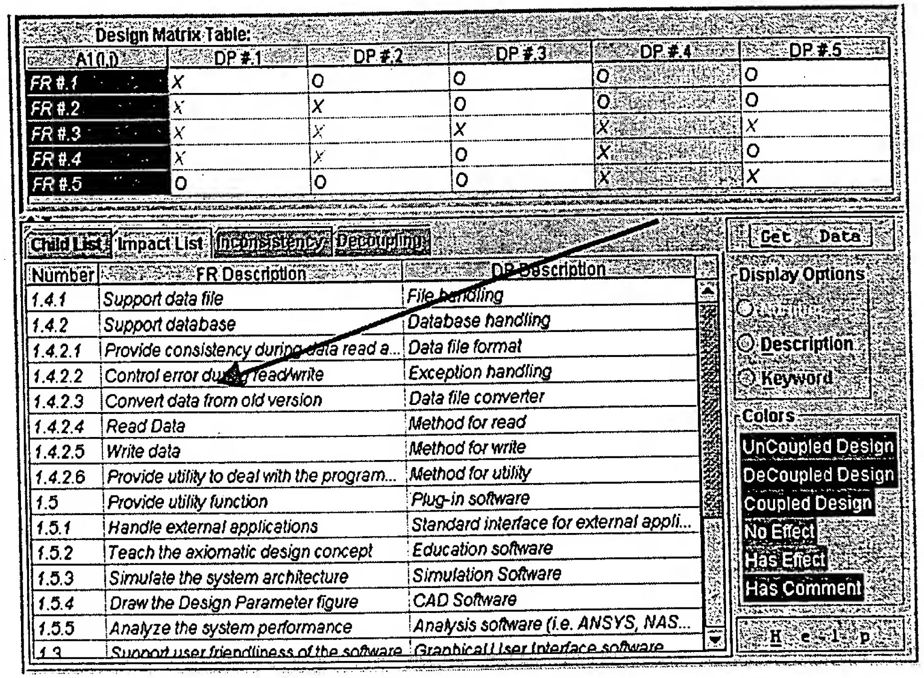

Design Matrix Table:
A1(i,j) | DP #1 | DP #2 | DP #3 | DP #4 | DP #5
FR #1 | X | O | O | O | O
FR #2 | X | X | O | O | O
FR #3 | X | X | X | X | X
FR #4 | X | X | O | X | O
FR #5 | O | O | O | X | X
Child List
Impact List
Inconsistency
Decoupling
Number | FR Description | DP Description
1.4.1 | Support data file | File handling
1.4.2 | Support database | Database handling
1.4.2.1 | Provide consistency during data read a... | Data file format
1.4.2.2 | Control error during read/write | Exception handling
1.4.2.3 | Convert data from old version | Data file converter
1.4.2.4 | Read Data | Method for read
1.4.2.5 | Write data | Method for write
1.4.2.6 | Provide utility to deal with the program... | Method for utility
1.5 | Provide utility function | Plug-in software
1.5.1 | Handle external applications | Standard interface for external appli...
1.5.2 | Teach the axiomatic design concept | Education software
1.5.3 | Simulate the system architecture | Simulation Software
1.5.4 | Draw the Design Parameter figure | CAD Software
1.5.5 | Analyze the system performance | Analysis software (i.e. ANSYS, NAS...
Get Data
Display Options
Description
Keyword
Colors
UnCoupled Design
DeCoupled Design
Coupled Design
No Effect
Has Effect
Has Comment
Help

FIGURE 74

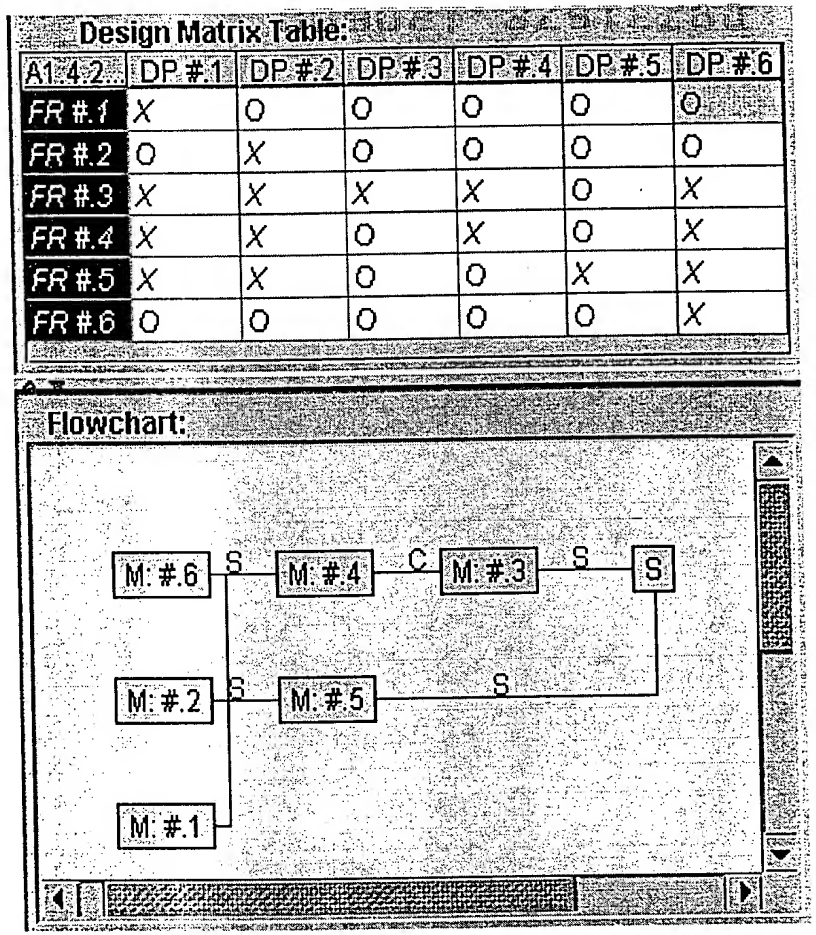

Design Matrix Table:
A1: 4.2
DP #.1
DP #.2
DP #.3
DP #.4
DP #.5
DP #.6
FR #.1 X O O O O O
FR #.2 O X O O O O
FR #.3 X X X X O X
FR #.4 X X O X O X
FR #.5 X X O O X X
FR #.6 O O O O O X
Flowchart:
M: #.6
S
M: #.4
C
M: #.3
S
S
M: #.2
S
M: #.5
S
M: #.1

FIGURE 75

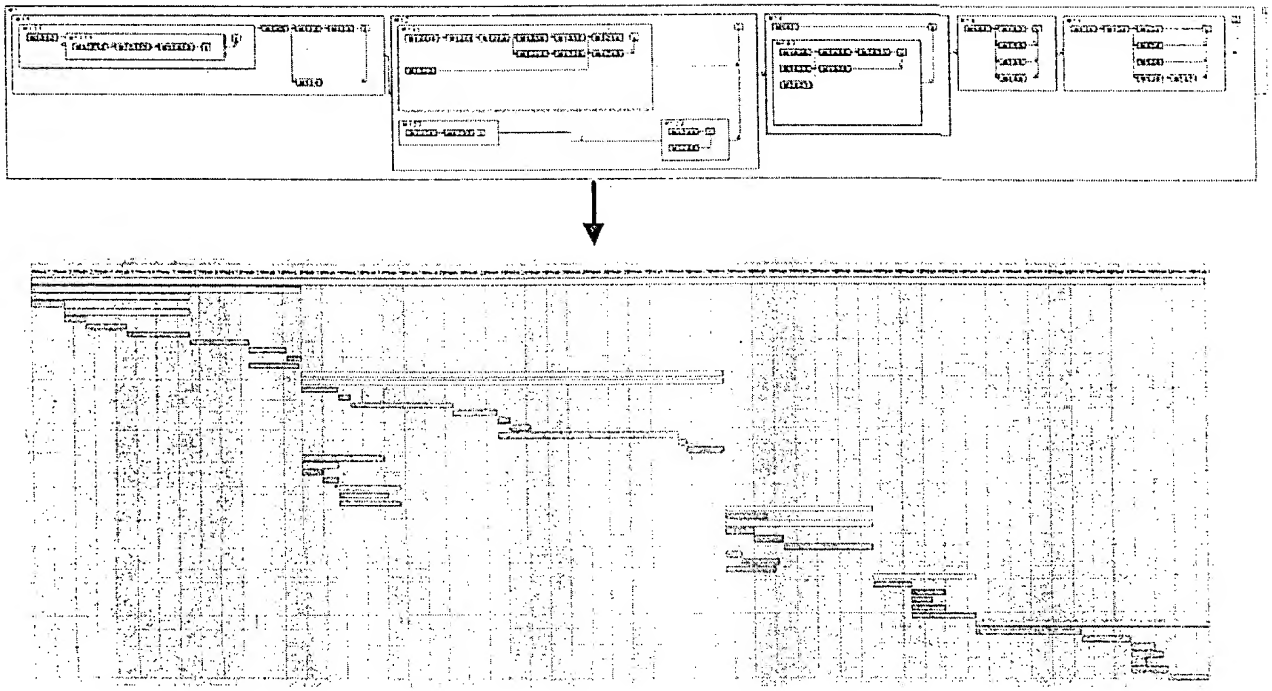


FIGURE 76

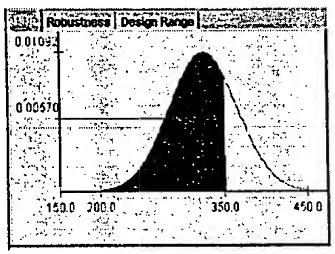


FIGURE 77A

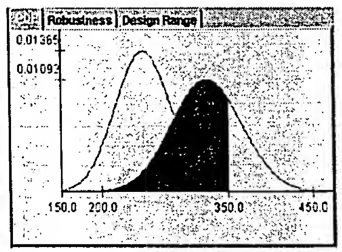


FIGURE 77B

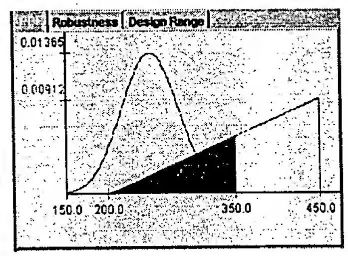


FIGURE 77C

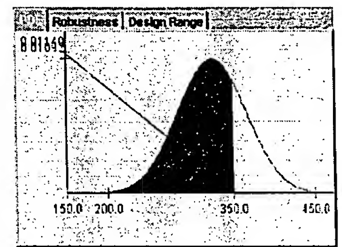
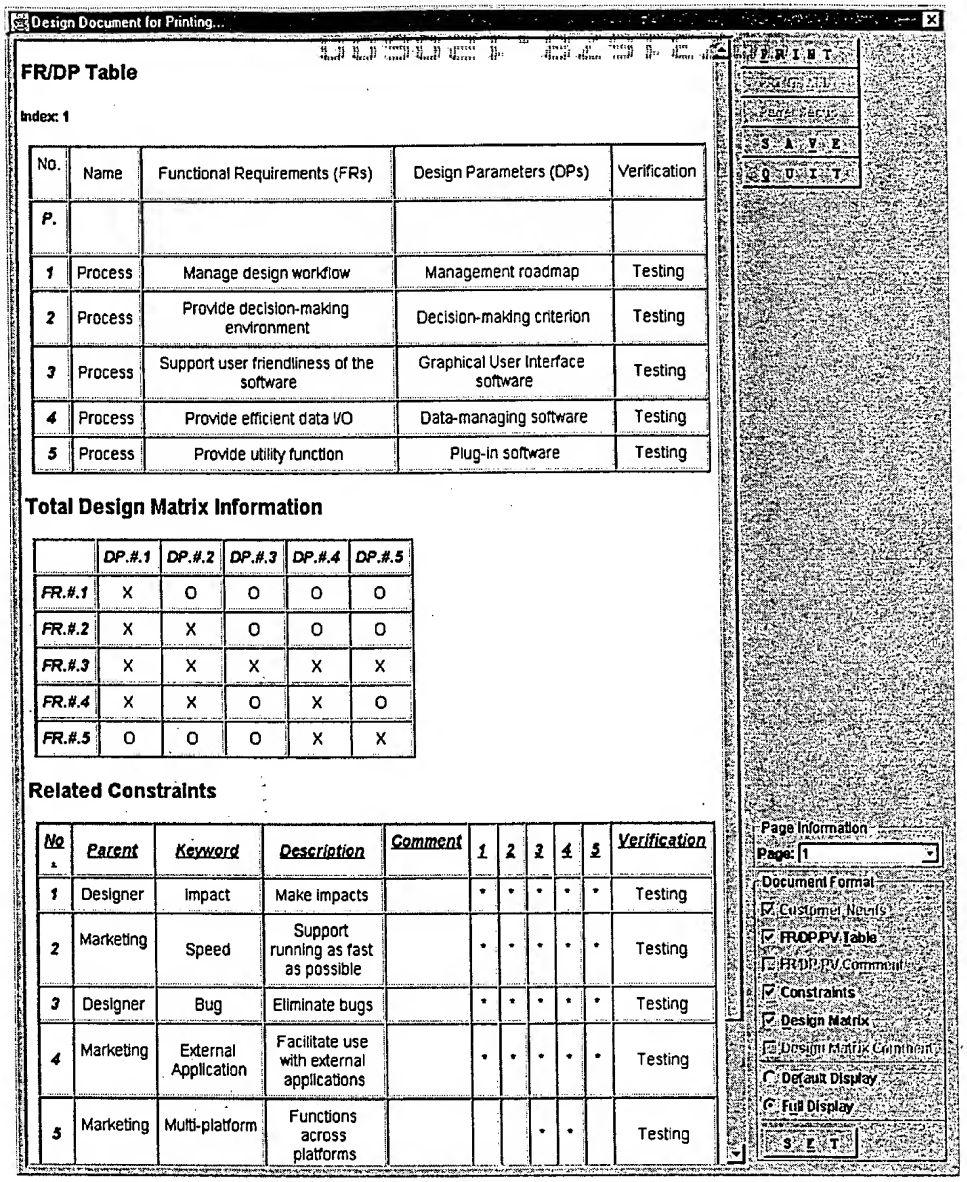


FIGURE 77D



No.	Name	Functional Requirements (FRs)	Design Parameters (DPs)	Verification
P.				
1	Process	Manage design workflow	Management roadmap	Testing
2	Process	Provide decision-making environment	Decision-making criterion	Testing
3	Process	Support user friendliness of the software	Graphical User Interface software	Testing
4	Process	Provide efficient data I/O	Data-managing software	Testing
5	Process	Provide utility function	Plug-in software	Testing

	DP.#.1	DP.#.2	DP.#.3	DP.#.4	DP.#.5
FR.#.1	X	O	O	O	O
FR.#.2	X	X	O	O	O
FR.#.3	X	X	X	X	X
FR.#.4	X	X	O	X	O
FR.#.5	O	O	O	X	X

No.	Parent	Keyword	Description	Comment	1	2	3	4	5	Verification
1	Designer	Impact	Make impacts		*	*	*	*	*	Testing
2	Marketing	Speed	Support running as fast as possible		*	*	*	*	*	Testing
3	Designer	Bug	Eliminate bugs		*	*	*	*	*	Testing
4	Marketing	External Application	Facilitate use with external applications		*	*	*	*	*	Testing
5	Marketing	Multi-platform	Functions across platforms				*	*		Testing

FIGURE 78

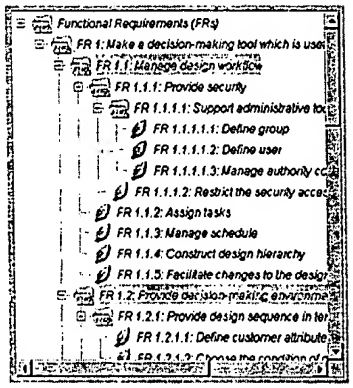


FIGURE 79A

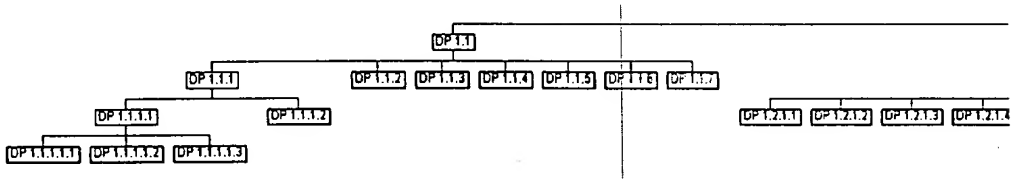


FIGURE 79B

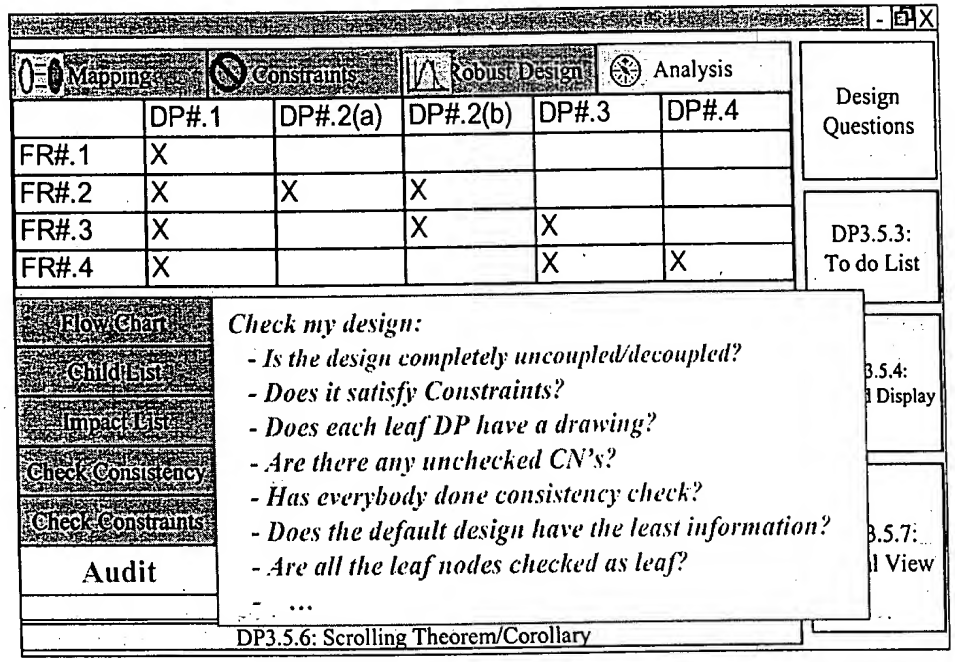


FIGURE 80

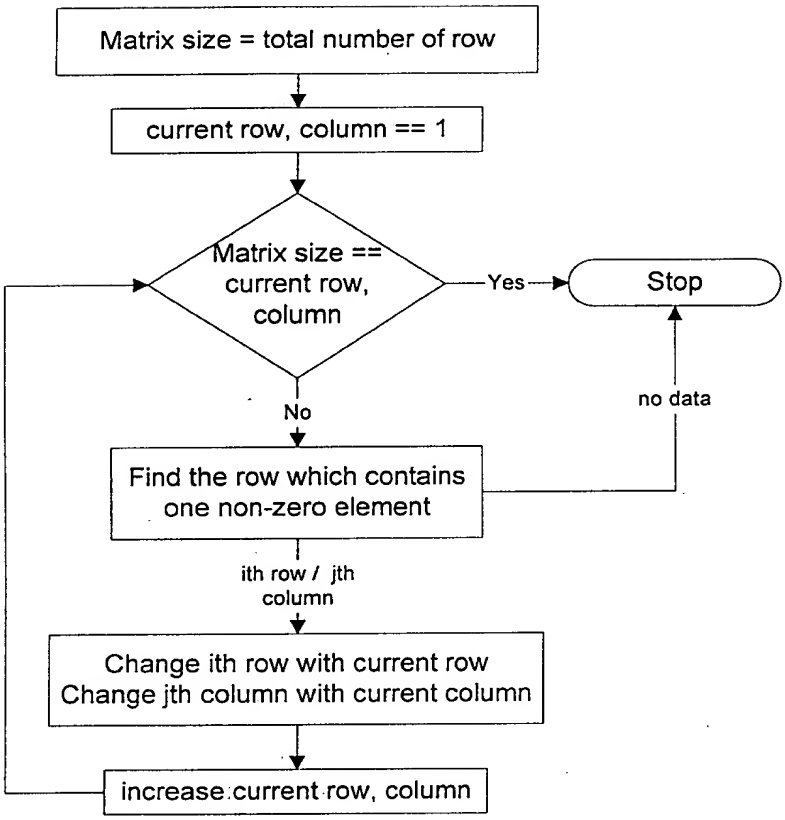

Matrix size = total number of row
current row, column == 1
Matrix size == current row, column
Yes
Stop
No
no data
Find the row which contains one non-zero element
ith row / jth column
Change ith row with current row
Change jth column with current column
increase current row, column

FIGURE 81

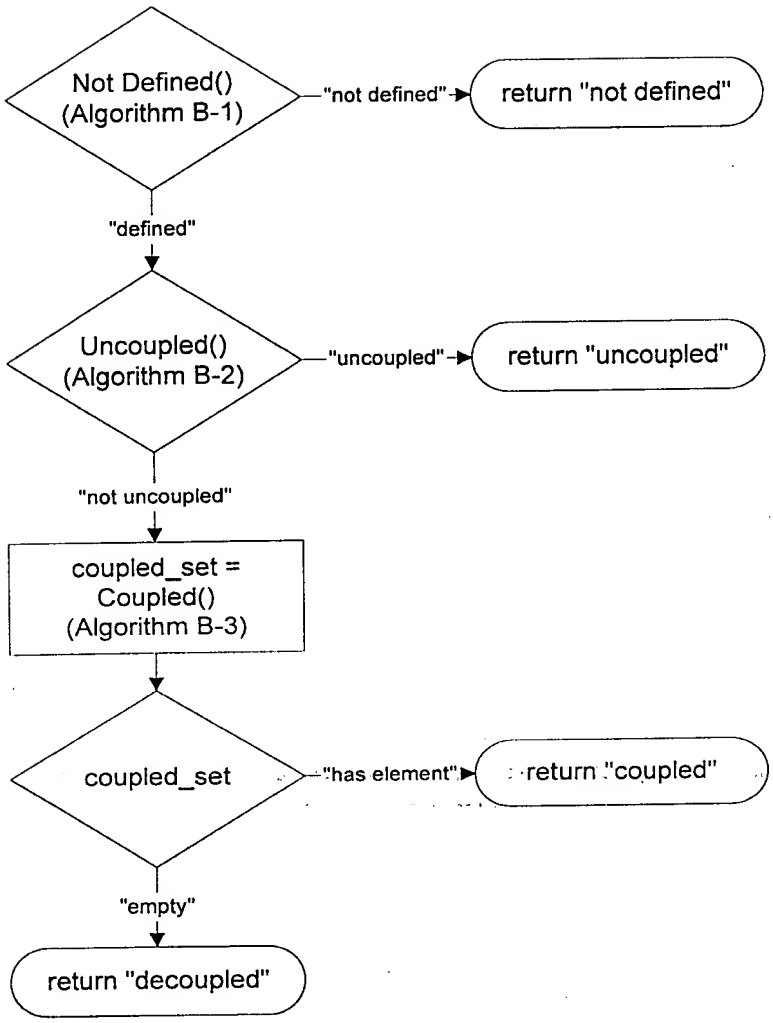

Not Defined()
(Algorithm B-1)
"not defined"
return "not defined"
"defined"
Uncoupled()
(Algorithm B-2)
"uncoupled"
return "uncoupled"
"not uncoupled"
coupled_set =
Coupled()
(Algorithm B-3)
coupled_set
"has element"
return "coupled"
"empty"
return "decoupled"

FIGURE 82

```
Loop One (int row=0; row<total_row_number: row++) {
  Loop Two (int column=0; column <total_column_number; column++) {
    If(maxtrix[row][column] == "empty")
      return "not defined"

    If(row == column) {
      If(matrix[row][column] == "O")
        return "not defined"
    }
  }
}

return "defined"
```

If one of the diagonal element has "0", the design is not defined in terms of the axiomatic design viewpoint

FIGURE 83

```
Loop One (int row=0; row<total_row_number: row++) {
  Loop Two (int column=0; column <total_column_number; column++) {
    If(row != column) {
      If(matrix[row][column] == "X")
        return "not uncoupled"
    }
  }
}

return "uncoupled"
```

FIGURE 84

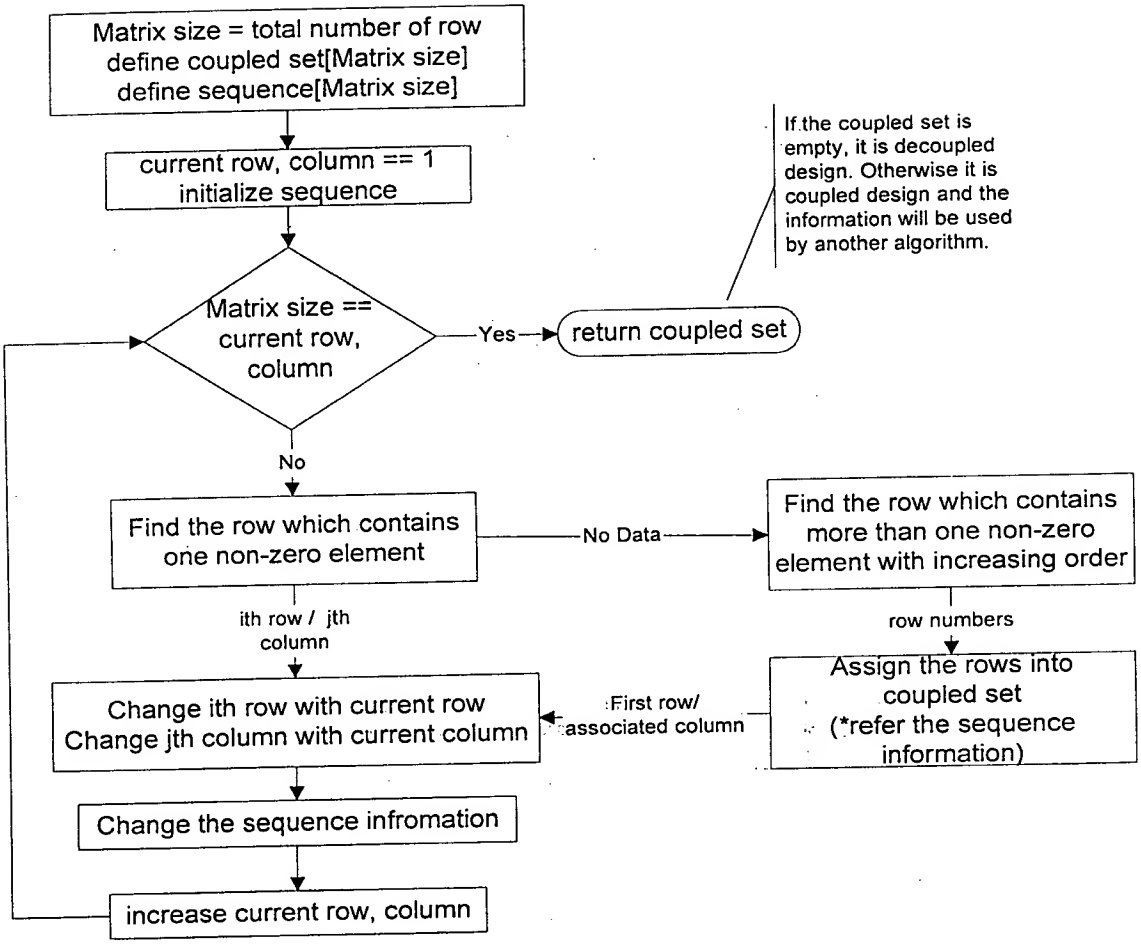

Matrix size = total number of row
define coupled set[Matrix size]
define sequence[Matrix size]
current row, column == 1
initialize sequence
Matrix size == current row, column
Yes
return coupled set
If the coupled set is empty, it is decoupled design. Otherwise it is coupled design and the information will be used by another algorithm.
No
Find the row which contains one non-zero element
No Data
Find the row which contains more than one non-zero element with increasing order
ith row / jth column
row numbers
Change ith row with current row
Change jth column with current column
First row/ associated column
Assign the rows into coupled set (*refer the sequence information)
Change the sequence infromation
increase current row, column

FIGURE 85

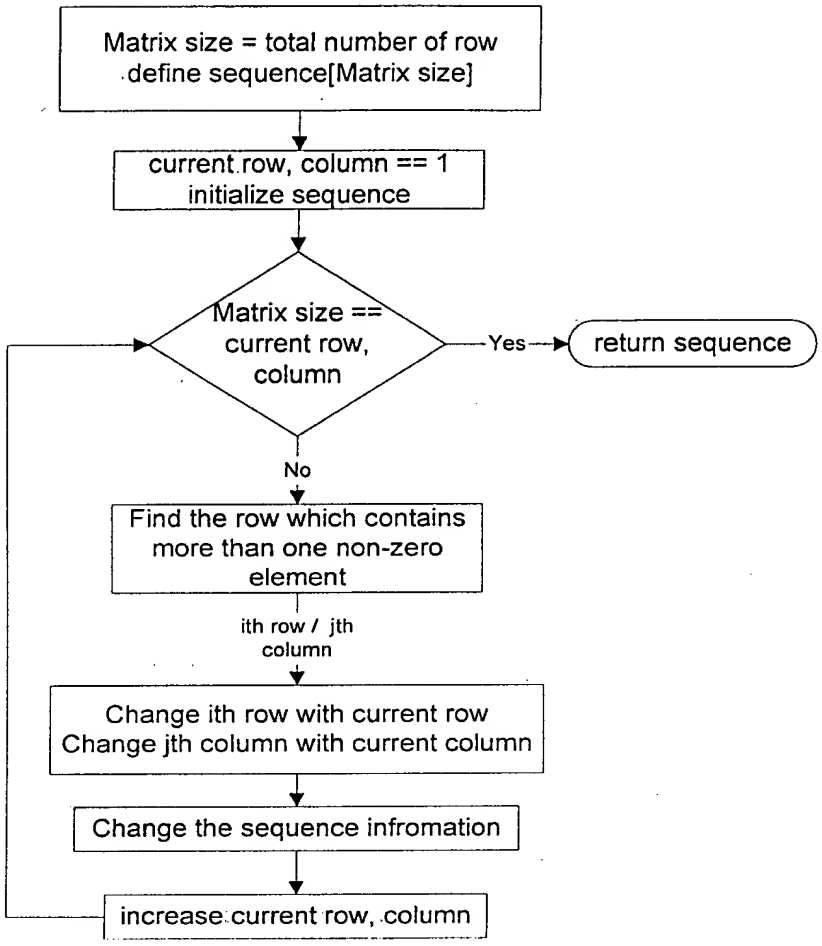
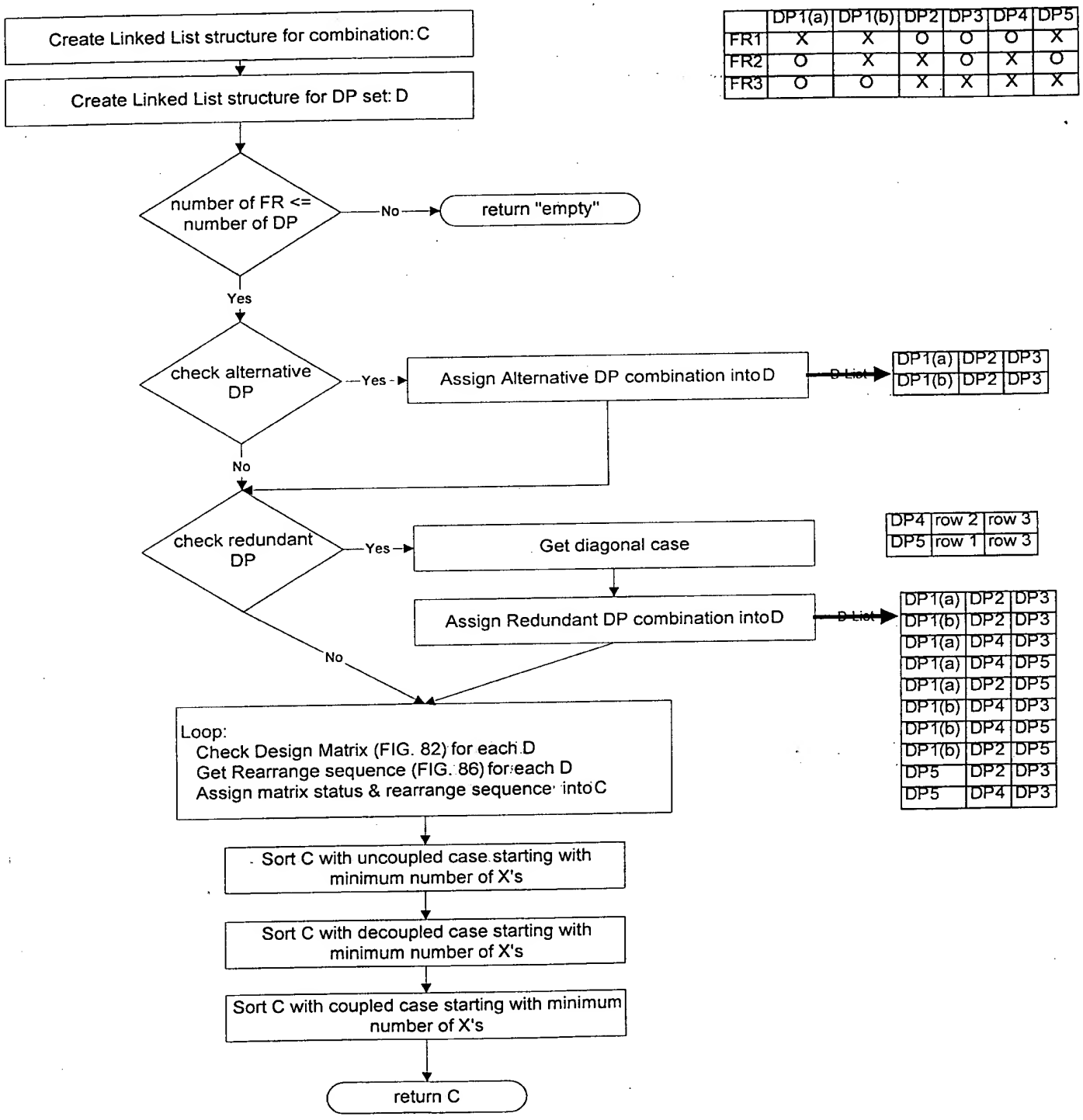

Matrix size = total number of row
define sequence[Matrix size]
current row, column == 1
initialize sequence
Matrix size == current row, column
Yes
return sequence
No
Find the row which contains more than one non-zero element
ith row / jth column
Change ith row with current row
Change jth column with current column
Change the sequence infromation
increase current row, column

FIGURE 86



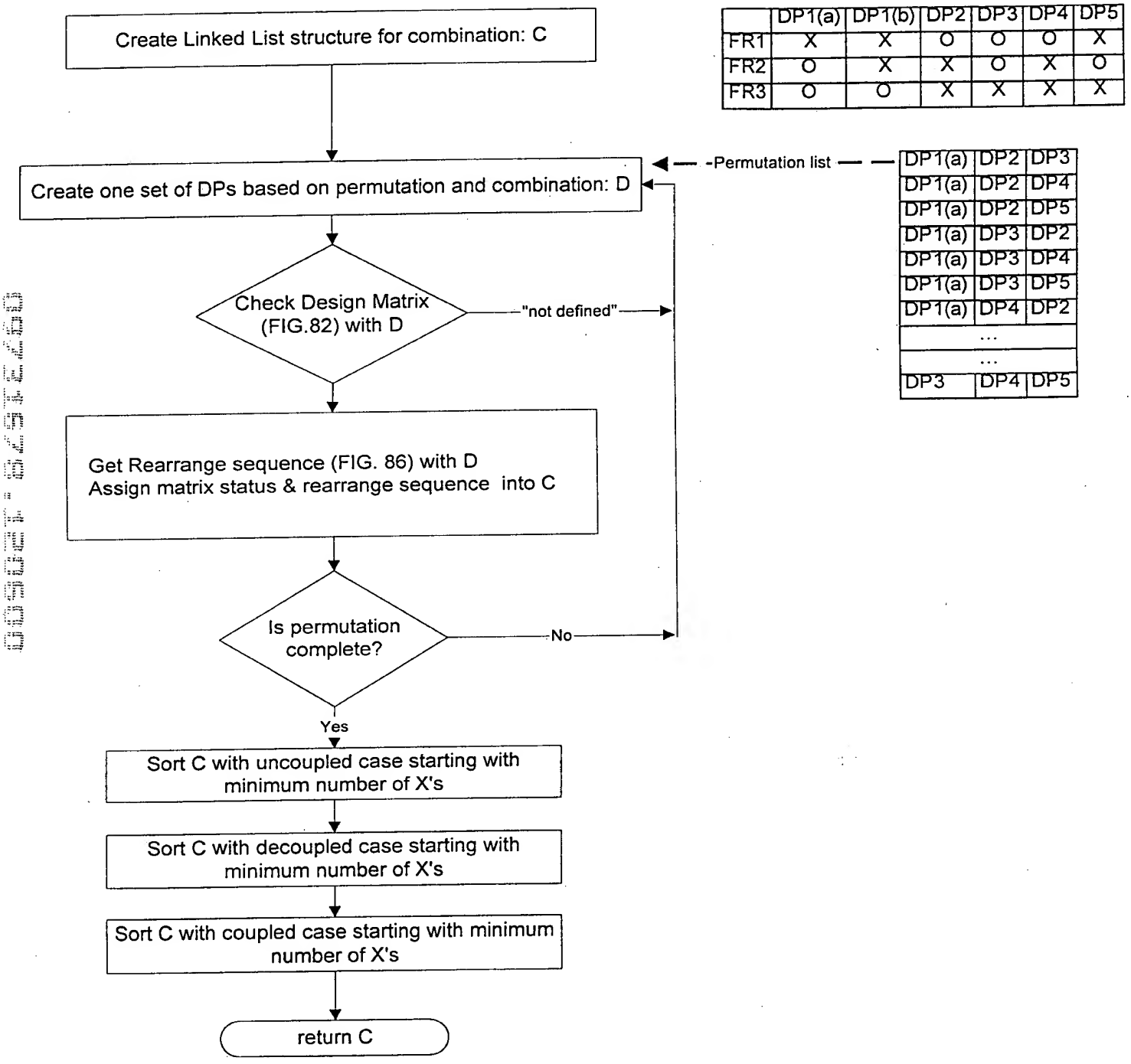
	DP1(a)	DP1(b)	DP2	DP3	DP4	DP5
FR1	X	X	O	O	O	X
FR2	O	X	X	O	X	O
FR3	O	O	X	X	X	X

DP1(a)	DP2	DP3
DP1(b)	DP2	DP3

DP4	row 2	row 3
DP5	row 1	row 3

DP1(a)	DP2	DP3
DP1(b)	DP2	DP3
DP1(a)	DP4	DP3
DP1(a)	DP4	DP5
DP1(a)	DP2	DP5
DP1(b)	DP4	DP3
DP1(b)	DP4	DP5
DP1(b)	DP2	DP5
DP5	DP2	DP3
DP5	DP4	DP3

FIGURE 87



	DP1(a)	DP1(b)	DP2	DP3	DP4	DP5
FR1	X	X	O	O	O	X
FR2	O	X	X	O	X	O
FR3	O	O	X	X	X	X

FIGURE 88

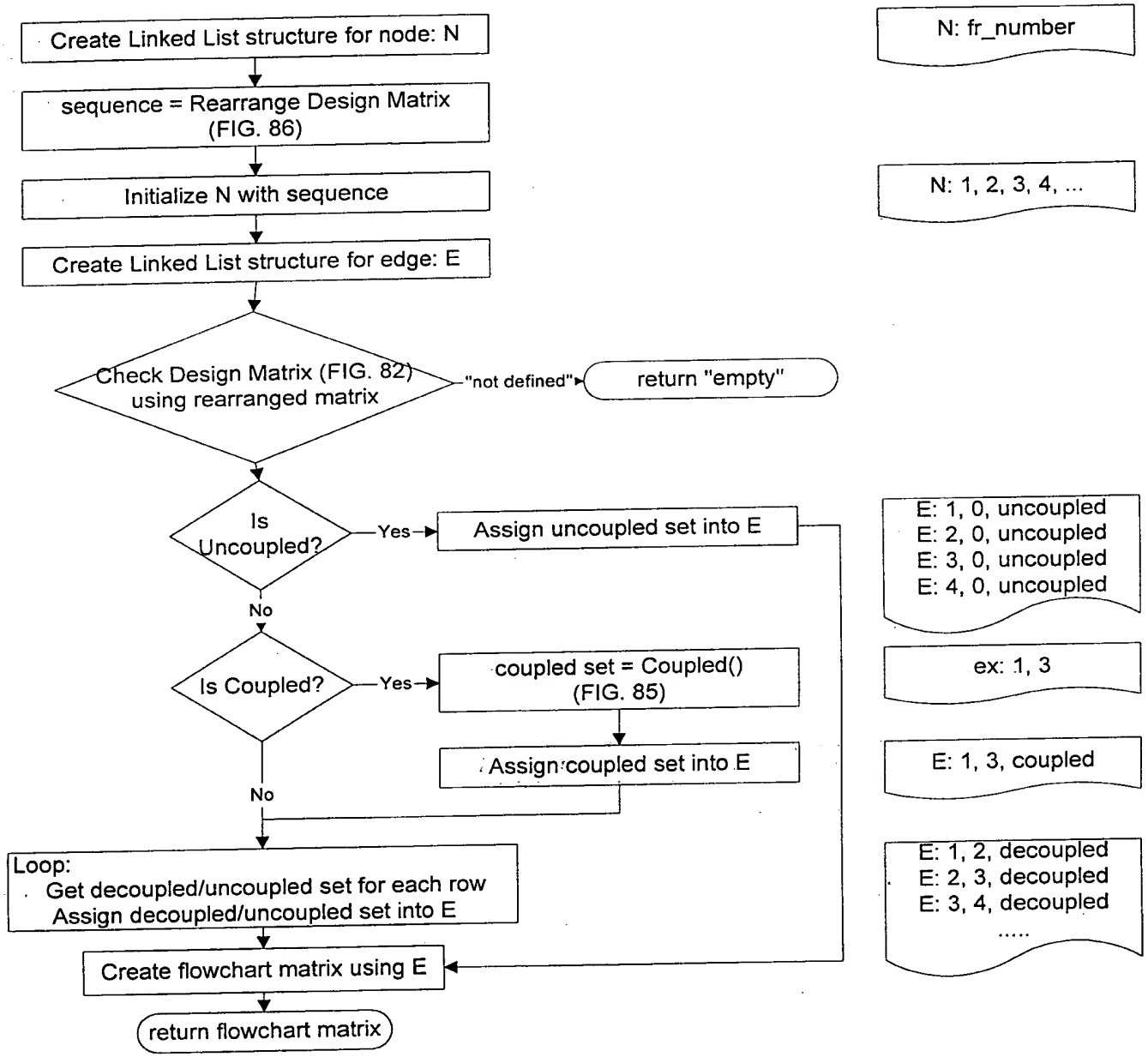

Create Linked List structure for node: N
sequence = Rearrange Design Matrix (FIG. 86)
Initialize N with sequence
Create Linked List structure for edge: E
Check Design Matrix (FIG. 82) using rearranged matrix
"not defined"
return "empty"
Is Uncoupled?
Yes
Assign uncoupled set into E
No
Is Coupled?
Yes
coupled set = Coupled() (FIG. 85)
Assign coupled set into E
No
Loop:
Get decoupled/uncoupled set for each row
Assign decoupled/uncoupled set into E
Create flowchart matrix using E
return flowchart matrix
N: fr_number
N: 1, 2, 3, 4, ...
E: 1, 0, uncoupled
E: 2, 0, uncoupled
E: 3, 0, uncoupled
E: 4, 0, uncoupled
ex: 1, 3
E: 1, 3, coupled
E: 1, 2, decoupled
E: 2, 3, decoupled
E: 3, 4, decoupled
.....

FIGURE 89

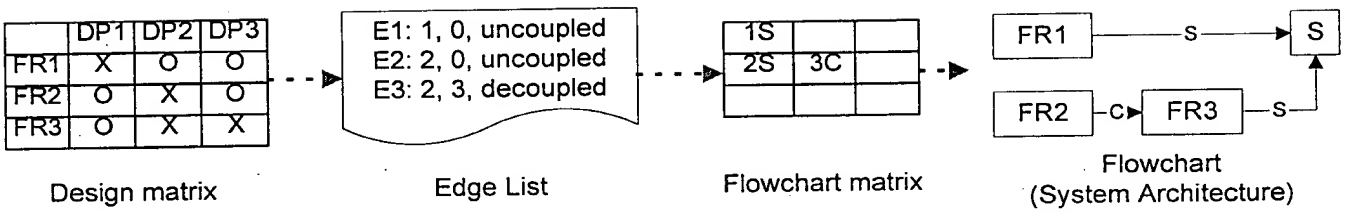

DP1
DP2
DP3
FR1
X
O
O
FR2
O
X
O
FR3
O
X
X
Design matrix
E1: 1, 0, uncoupled
E2: 2, 0, uncoupled
E3: 2, 3, decoupled
Edge List
1S
2S
3C
Flowchart matrix
FR1
S
S
FR2
C
FR3
S
Flowchart
(System Architecture)

FIGURE 90

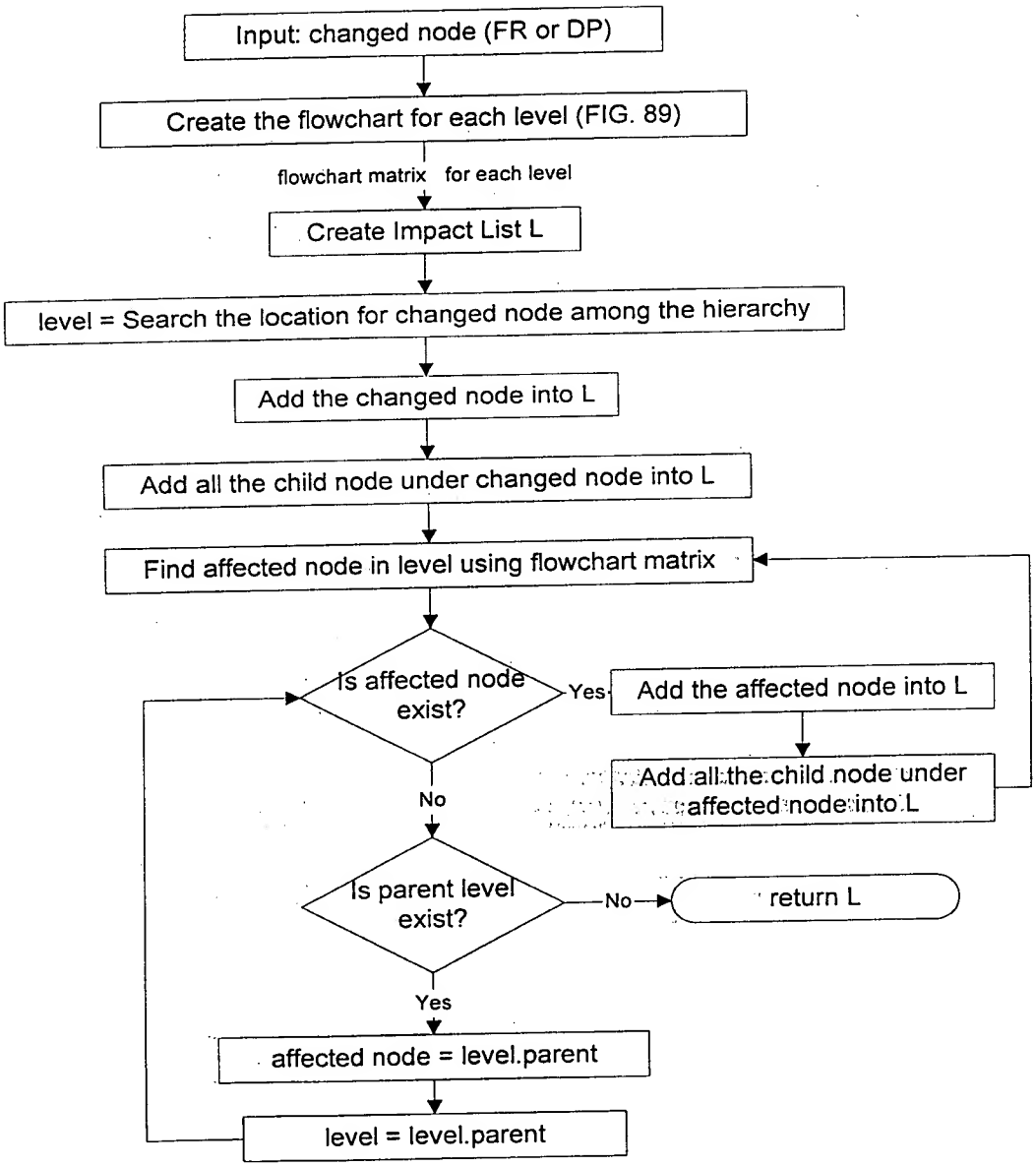

Input: changed node (FR or DP)
Create the flowchart for each level (FIG. 89)
flowchart matrix for each level
Create Impact List L
level = Search the location for changed node among the hierarchy
Add the changed node into L
Add all the child node under changed node into L
Find affected node in level using flowchart matrix
Is affected node exist?
Yes
Add the affected node into L
Add all the child node under affected node into L
No
Is parent level exist?
No
return L
Yes
affected node = level.parent
level = level.parent

FIGURE 91

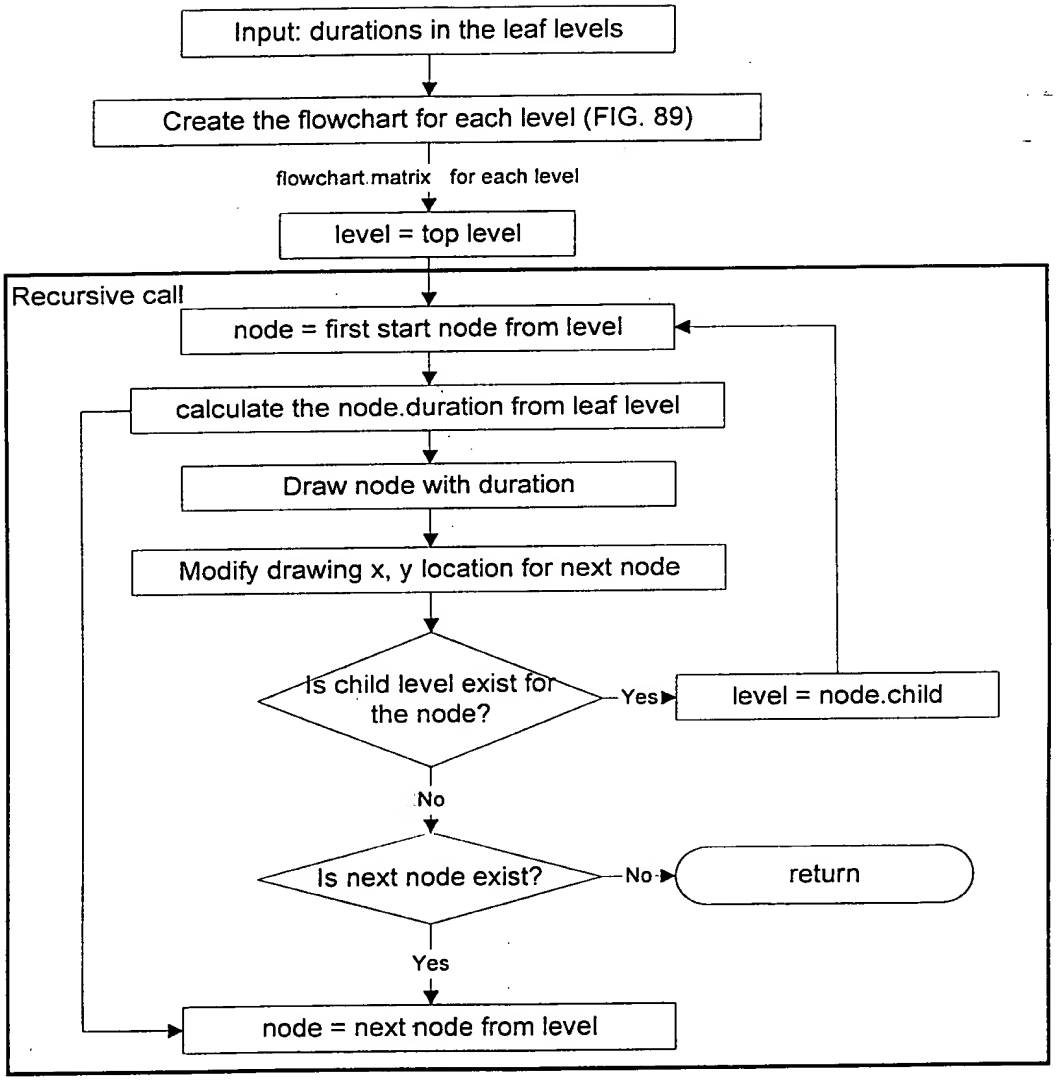

Input: durations in the leaf levels
Create the flowchart for each level (FIG. 89)
flowchart.matrix for each level
level = top level
Recursive call
node = first start node from level
calculate the node.duration from leaf level
Draw node with duration
Modify drawing x, y location for next node
Is child level exist for the node?
Yes
level = node.child
No
Is next node exist?
No
return
Yes
node = next node from level

FIGURE 92